

Una propuesta didáctica en 3º E.S.O para trabajar el pensamiento matemático avanzado haciendo uso de Scratch

Miguel Ángel Baeza-Alba y Francisco Javier Claros-Mellado
Universidad Complutense de Madrid
M^a Teresa Sánchez-Campaña
Universidad de Málaga

RESUMEN: *En este trabajo se presenta una propuesta didáctica llevada a cabo con alumnos de 3º de E.S.O, basada en la programación del Algoritmo de Euclides para el Máximo Común Divisor, con la herramienta tecnológica Scratch. Esta propuesta permitió trabajar, elementos propios del Pensamiento Matemático Avanzado (PMA) como son la abstracción, la formalización y la generalización a través de una metodología basada en la Teoría de las Situaciones Didácticas (TSD). Se observó cómo el diseño de las fichas de trabajo, siguiendo la TSD, junto con la organización de la clase en grupos, favoreció el debate y la obtención de resultados.*

PALABRAS CLAVE: *Algoritmo, Scratch, Máximo Común Divisor, Pensamiento Matemático Avanzado, Teoría de las Situaciones Didácticas.*

A methodological approach in 3rd E.S.O to work the advanced mathematical thinking using Scratch

ABSTRACT: *In this document we will carry out a methodological approach with the students from 3º E.S.O based on programming Euclides Algorithm for the Greatest Common Divisor (MCD) with Scratch technological tool. This proposal allows working elements of Advanced Mathematical Thinking (PMA) such as abstraction, formalization and generalization. The methodology followed is based on the Theory of Didactic Situations (TSD). In addition, the design of the worksheets, following the TSD, along with the organization of the class into groups, favored the debate and achieving results.*

KEYWORDS: *Algorithm, Scratch, Greatest Common Divisor, Advanced Mathematical Thinking, Theory of Didactic Situations*

INTRODUCCIÓN

Las nuevas tecnologías de la información y la comunicación se definen según Adell (1997, p.7) como: “... el conjunto de procesos y productos derivados de las nuevas herramientas (*hardware y software*), soportes de la información y canales de comunicación relacionados con el almacenamiento, procesamiento y transmisión digitalizados de los datos”.

La sociedad está cambiando y nuestros alumnos con ella. Cada vez más, las nuevas tecnologías de la información están en todos los ámbitos de nuestra vida y, por tanto, deben estar también en la escuela. Este hecho va a llevar emparejado una serie de cambios; Pérez (1998) sugirió reconceptualizar el alcance de lo educativo, reformular el currículo e innovar en las estrategias educativas.

Pero es claro que, para hacer un buen uso de la tecnología en el aula, el profesorado necesita formación en nuevas metodologías de trabajo que le permitan adaptarse al nuevo entorno en el que se encuentra. Pascual (1998) advierte que el uso del ordenador no supone una mejora si no va acompañado de un adecuado planteamiento metodológico en el que se deben incluir los objetivos y contenidos a trabajar y el software más adecuado para hacerlo; el uso que hagamos del ordenador en el aula, tiene que formar parte de una actividad que haya sido diseñada previamente por el profesor. Para llevar a cabo este planteamiento pedagógico, los profesores deben tener una sólida formación en las herramientas que se van a manejar.

Ricoy y Couto (2012) señalan los beneficios, controversias y mejoras que puede aportar la introducción de las TIC en el aula. Entre las dificultades en su utilización señalan como importantes la desmotivación de ciertos docentes ante su uso y la escasez de medios tecnológicos en algunos centros. Por el contrario, aseguran que ayuda a atender a la diversidad y favorece la comunicación bidireccional.

Han surgido muchos programas en los últimos años para trabajar las matemáticas, que pretenden ser una herramienta útil en la enseñanza de las mismas. Fernández y Muñoz (2007) hacen un recorrido sobre los programas matemáticos que más suelen emplearse en el aula. Mencionan, entre otros, Wiris, Geogebra, Cabri y Derive, así como páginas web que aportan información y ejercicios para trabajar las matemáticas. No hemos encontrado entre las actividades propuestas, ninguna relacionada con la programación.

Scratch es un software libre desarrollado por Lifelong Kindergarten Group de los Laboratorios Media-Lab en MIT. Es un lenguaje gráfico que permite programar uniendo bloques predefinidos. Su fácil manejo hace que pueda iniciarse en él a edades muy tempranas. En la actualidad, hay muchas funcionalidades que pueden aplicarse a Scratch y que son objeto de investigación. Carralero (2011) señala como funcionalidad importante el hecho de manejar la programación implícita en Scratch para trabajar un contenido de primaria y secundaria. Asimismo señala que para abordar determinados elementos de la programación (por ejemplo los bucles), es necesario que los alumnos se encuentren al menos en 3º ESO, ya que a esa edad éstos empiezan a poseer un pensamiento lógico-abstracto. En la construcción del programa a desarrollar, puede ser necesaria la resolución de algoritmos, elementos muy importantes en el desarrollo de las matemáticas.

De acuerdo con su origen, el término algoritmo debería hacer referencia a las distintas técnicas calculistas surgidas en el campo aritmético. No obstante, al haber sido

traspasados en la actualidad los límites de la aritmética, se pueden encontrar definiciones como las siguientes:

“Un algoritmo es una sucesión finita de reglas elementales, regidas por una prescripción precisa y uniforme, que permite efectuar paso a paso, en un encadenamiento estricto y riguroso, ciertas operaciones de tipo ejecutable, con vistas a la resolución de los problemas pertenecientes a una misma clase” (Ifrah, 2008, p. 1616)

“Un algoritmo es una prescripción, una orden o sistema secuenciado de órdenes que encadenan una serie de operaciones elementales que llevan desde los datos iniciales al resultado” (Gairín y Sancho, 2002, p. 83)

Si las Matemáticas tienen como objetivo prioritario resolver problemas y encontrar soluciones a cuestiones cada vez más difíciles, parece que la necesidad de utilizar algoritmos está totalmente justificada.

Usiskin (1998), por su parte, enumera hasta nueve razones diferentes por las que es útil saber y enseñar algoritmos matemáticos: son eficaces, fiables, precisos, rápidos, proporcionan un registro escrito, establecen una imagen mental, son instructivos, pueden ser utilizados en otros algoritmos y pueden ser objetos de estudio (comparación de eficacia, características matemáticas, análisis de velocidad, entre otras características).

En este documento llevaremos a cabo una propuesta didáctica que pretende trabajar el Pensamiento Matemático Avanzado, en el sentido de Tall (1991), con alumnos 3º E.S.O en la asignatura Matemáticas para las Enseñanzas Académicas. Para ello se trabajará el Algoritmo de Euclides para el cálculo del M.C.D, utilizando la herramienta Scratch.

El carácter iterativo del algoritmo de Euclides para calcular el M.C.D, es una de las razones principales por las que consideramos que este concepto puede ser tratado a partir de la herramienta tecnológica Scratch, proporcionando un aprendizaje significativo del concepto e introduciendo a los alumnos en los procesos cognitivos propios del Pensamiento Matemático Avanzado. Marmolejo y Campos (2012) señalan la importancia de la introducción del Scratch incluso en primaria, ya que puede hacer que los alumnos desarrollen capacidades intelectuales de orden superior como son el análisis, la síntesis, la conceptualización, etc. Estas capacidades prepararán sin lugar a duda para el desarrollo de conceptos de una dificultad superior.

El documento se organiza en torno a seis apartados. El primero es la presente introducción. En el segundo señalamos los objetivos de la investigación. En el tercero, denominado marco teórico, describiremos dos teorías en las que nos apoyaremos: el Pensamiento Matemático Avanzado y la Teoría de las Situaciones Didácticas. En el cuarto describimos la metodología que vamos a utilizar. En el quinto describimos los resultados obtenidos en la ficha tres y por último, en el sexto, señalamos las principales conclusiones obtenidas, así como las perspectivas futuras que esperamos abordar.

OBJETIVOS

El DECRETO 48/2015, de 14 de mayo, del Consejo de Gobierno, por el que se establece para la Comunidad de Madrid el currículo de la Educación Secundaria Obligatoria,

establece para 3ºESO en el apartado dedicado a las matemáticas para las enseñanzas académicas, las siguientes recomendaciones:

- *Planificación en el proceso de resolución de problemas en la que se desarrollen estrategias de resolución y reflexión sobre los resultados obtenidos.*
- *Planteamiento de investigaciones matemáticas en las que se pongan en práctica los procesos de matematización y modelización.*
- *Utilización de los medios tecnológicos en el proceso de aprendizaje.*

En esta investigación tendremos en cuenta estos tres elementos para resolver el problema siguiente: “Diseñar e implementar un algoritmo en Scratch que permita calcular el MCD de dos números naturales”.

La resolución de este problema lleva implícito el uso de unas formas de pensamiento denominadas Pensamiento Matemático Avanzado, que son propias de una matemática superior. Entre los elementos que forman parte de este tipo de pensamiento se encuentran: abstracción, generalización, formulación de hipótesis, modelización y desarrollo lógico de un proceso. Estos serán los elementos que pretendemos que los alumnos desarrollen y que tradicionalmente no están presentes en la enseñanza secundaria. Esto último, según nuestra experiencia como docentes, puede ser debido a temarios muy extensos que llevan más a la enseñanza de procedimientos que a la enseñanza de conceptos y aplicaciones de los mismos.

Los objetivos que pretendemos conseguir en esta investigación son los siguientes:

- O1. Iniciar a los alumnos en el razonamiento lógico-matemático a través del diseño del algoritmo del MCD en 3ºESO.
- O2. Potenciar en los alumnos el uso de la abstracción a través de la traducción a Scratch del algoritmo diseñado. Es decir, potenciar la abstracción a través de la implementación del algoritmo para el MCD en Scratch.
- O3. Formular hipótesis que permitan ser contrastadas a través del algoritmo diseñado e implementado.
- O4. Introducir en los alumnos el concepto de generalización a partir de la comprobación de casos particulares en la aplicación creada (números primos entre sí, números uno múltiplo del otro, números iguales, etc.)
- O5. Valorar el concepto de modelo matemático.

Para conseguir estos objetivos nos apoyaremos en el marco teórico que describimos a continuación.

MARCO TEÓRICO

Nuestro marco teórico se sustenta en dos teorías didácticas. La primera teoría, Pensamiento Matemático Avanzado (PMA), se usará para determinar los procesos implicados en el diseño e implementación del algoritmo de Euclides para calcular el MCD con Scratch. Nos basaremos en ella para elegir el contenido y la secuenciación de las actividades. A su vez, éstas estarán diseñadas siguiendo la Teoría de Situaciones Didácticas.

Pensamiento Matemático Avanzado (PMA)

En 1985, en el seno del grupo Psychologist Mathematic Education (PME), se formó un grupo de trabajo denominado Pensamiento Matemático Avanzado. El objetivo de dicho grupo era investigar los procesos implicados en la construcción de conceptos que por su naturaleza y dificultad formaban parte de la matemática superior. A partir de ese momento, el interés en didáctica de la matemática se empieza a centrar en la problemática del aprendizaje en términos de procesos cognitivos y no como una simple adquisición de competencias y habilidades; además, se produjo una evolución en las investigaciones que se empezaron a preocupar de tópicos que pertenecen a una “matemática superior”, tales como los límites, las derivadas o las integrales.

Entre los procesos involucrados en el PMA, los más importantes son la abstracción y la generalización. Dreyfus (1991) define la abstracción como un proceso de construcción de objetos mentales a partir de objetos matemáticos y la generalización como la derivación e inducción de particulares para identificar generalidades y extender dominio de validez.

Señalamos las dificultades para distinguir la línea que separa el Pensamiento Matemático Elemental (PME) del Pensamiento Matemático Avanzado (PMA). Tall (1985; 1991) afirma que el paso del PME al PMA implica una transición significativa que requiere una reconstrucción cognitiva. Esta reconstrucción implica pasar del “describir” a “definir” y pasar del “convencer” a “demostrar”.

Robert y Swarzenberger (1991) señalaron una serie de diferencias entre el pensamiento elemental y el avanzado:

- En el Pensamiento Matemático Avanzado los alumnos tienen que aprender más conceptos en menos tiempo y además éstos son presentados de manera formal.
- Los conceptos enseñados llevan asociadas las siguientes propiedades: generalización, abstracción y formalización; propiedades que pueden entrar en conflicto con el conocimiento anterior que se tenía sobre el concepto.
- Los alumnos se enfrentan a una amplia gama de problemas que nacen de una variedad de contextos, los cuales no pueden ser discutidos en todo detalle.

Estas diferencias entre el Pensamiento Matemático Elemental y el Pensamiento Matemático Avanzado propuestas por Tall (1985; 1991) y Robert y Swarzenberger (1991) fueron rebatidas por Edwards, Dubinsky y McDonald (2005); los cuales, además de proponer una definición alternativa de Pensamiento Matemático Avanzado, señalan que un concepto se considerará dentro del Pensamiento Matemático Avanzado dependiendo de los aspectos de este que se traten.

Edwards, Dubinsky y McDonald (2005; pp.17-18) proponen la siguiente definición de Pensamiento Matemático Avanzado:

“Pensamiento que requiere deductivo y riguroso razonamiento acerca de nociones matemáticas que no nos son enteramente accesibles a través de los cinco sentidos”.

Teniendo en cuenta todo lo anterior, si se realiza un tratamiento procedimental de un concepto, dicho concepto quedará fuera del Pensamiento Matemático Avanzado, a pesar de que sea uno que por su naturaleza debiera formar parte de él.

Teoría de las Situaciones Didácticas (TSD)

La Teoría de las Situaciones Didácticas (TSD) apareció en 1970. Nació como un método simple de descripción y de interrogación matemática de los dispositivos psicológicos y didácticos. Es una teoría para la enseñanza de las matemáticas que se basa en una concepción constructivista del aprendizaje. Según Brousseau (1986; p. 11):

“El alumno aprende adaptándose a un medio que es factor de contradicciones, de dificultades, de desequilibrios, un poco como lo hace la sociedad humana. Este saber, fruto de la adaptación del alumno, se manifiesta por respuestas nuevas que son la prueba del aprendizaje.”

Según Brousseau (2011), la teoría de las situaciones didácticas modeliza las condiciones bajo las cuales los seres humanos producen, comunican y asimilan los conocimientos matemáticos. Estas condiciones son modelizadas por sistemas llamados “situaciones”, que conducen a agentes en interacción con ellas a manifestar este conocimiento. Son pues específicas del conocimiento en juego.

Algunas situaciones requieren de un conocimiento previo para poder tener éxito en alcanzar el nuevo conocimiento, pero hay otras situaciones en las que el sujeto puede construir por sí mismo el conocimiento requerido sin recurrir a ningún conocimiento anterior.

Brousseau (1998) distingue, en el campo de la enseñanza de las matemáticas, dos tipos de situaciones: las didácticas y la a-didácticas (generalmente incluidas como una fase, dentro de las situaciones didácticas).

Panizza (2003; p.4) define de forma concisa y clara el concepto de situación didáctica:

“La situación didáctica es una situación construida intencionalmente con el fin de hacer adquirir a los alumnos un saber determinado.”

Brousseau (1986) describe el término de situación a-didáctica (o fase a-didáctica dentro de una situación didáctica) refiriéndose a toda situación en la que se hace necesario que el alumno ponga en práctica los conocimientos que se pretende que aprenda sin la intervención del profesor, y que además, la situación en sí misma tiene la capacidad de verificar si las decisiones que toma el alumno son acertadas o no.

Se distinguen tres tipos de situaciones (véase, Panizza (2003)):

- Situaciones de acción: el alumno debe actuar sobre un medio (material, o simbólico); la situación requiere solamente la puesta en acto de conocimientos implícitos.
- Situaciones de formulación: un alumno (o grupo de alumnos) emisor debe formular explícitamente un mensaje destinado a otro alumno (o grupo de alumnos) receptor que debe comprender el mensaje y actuar (sobre un medio, material o simbólico) en base al conocimiento contenido en el mensaje.
- Situaciones de validación: dos alumnos (o grupos de alumnos) deben enunciar aserciones y ponerse de acuerdo sobre la verdad o falsedad de las mismas. Las afirmaciones propuestas por cada grupo son sometidas a la consideración del otro

grupo, que debe tener la capacidad de “sancionarlas”, es decir ser capaz de aceptarlas, rechazarlas, pedir pruebas, oponer otras aserciones.

Otro concepto clave es el de institucionalización. Brousseau (1994) lo definió como la fase del proceso didáctico en la que, por un lado, el profesor verifica el aprendizaje del alumno, y por otro, el alumno verifica el objeto de enseñanza. En la institucionalización, el profesor realiza una revisión de las actividades realizadas y le da un estatus oficial. En ese momento, el contenido matemático pasa a formar parte del saber de la clase.

METODOLOGÍA

En este apartado describimos la metodología que se siguió a lo largo de toda la secuencia para cumplir con los objetivos propuestos.

Para ello se diseñaron siete fichas de trabajo que se llevaron a cabo con un grupo de 23 alumnos de 3º de la E.S.O de Matemáticas para las Enseñanzas Académicas del Colegio Concertado Liceo San Pablo de Leganés. Las fichas fueron realizadas por once equipos de dos alumnos cada uno y un último equipo de un único alumno. Los equipos fueron nombrados por el profesor, intentando que todos ellos estuviesen equilibrados. Al ser un número impar de alumnos, fue necesario formar un grupo de 3 o un grupo de 1. Finalmente se decidió formar un grupo de 1, situando en dicho grupo al alumno con mejores aptitudes para la tecnología. El tiempo dedicado a cada una de las fichas se describe a continuación. Las fichas de los diferentes grupos fueron recogidas por el profesor al acabar el tiempo fijado.

La ficha 0 tenía como objetivo familiarizar a los alumnos con el lenguaje de programación Scratch. Aunque los alumnos tenían ciertas nociones del mismo gracias a la asignatura de Tecnología, Programación y Robótica; no dominaban el uso de las variables y bucles, indispensables para nuestra secuencia. Por tanto, fue necesaria una introducción previa para no convertir la herramienta en un obstáculo para el aprendizaje. El tiempo dedicado para la misma fue de dos sesiones de 55 minutos cada una.

La ficha 1 tenía como objetivo introducir a los alumnos la noción de algoritmo, señalando la importancia que éstos tienen en matemáticas. La ficha se inició con la explicación del algoritmo de Euclides para el máximo común divisor. Se pusieron dos ejemplos sobre cómo llevarlo a la práctica y a continuación se pidió a los alumnos que realizaran 4 ejercicios de aplicación del mismo. El profesor creyó pertinente realizar los ejemplos resueltos en la pizarra, para así conseguir que todos los alumnos comprendiesen perfectamente el procedimiento de cálculo, ya que dicho algoritmo no era conocido previamente por los estudiantes. Durante la explicación del algoritmo, el profesor se ayudó de una tabla, que recogía en cada paso cual era el dividendo, divisor y resto. Para el caso del MCD de 150 y 90 (primer ejemplo explicado a los alumnos), la mencionada tabla fue la siguiente:

DIVIDENDO	DIVISOR	RESTO
150	90	60
90	60	30
60	30	0

Tabla 1. Tabla para la resolución del ejemplo 1 de la Ficha 1

El objetivo de la presente tabla fue el de hacer comprender a los alumnos cual era la condición de parada (resto 0) y hacerles comprender también cuál es el resultado que da el algoritmo al MCD (el último divisor). Cabe mencionar que dicha tabla constituyó una ayuda inestimable para la resolución de los ejercicios por los alumnos, ya que todos los grupos hicieron uso de ella para resolverlos. La duración asignada a esta ficha fue una sesión de 55 minutos.

La ficha 2 estaba encaminada a conseguir que los alumnos fueran capaces de diseñar diagramas de flujo, a partir de un determinado algoritmo. Estos diagramas son un instrumento útil y previo a la programación de una aplicación; no son más que una representación gráfica de los pasos que deberán seguir a la hora de programar el algoritmo. Con dicho diseño pretendemos iniciar a los alumnos en el razonamiento lógico matemático, además de potenciar la abstracción (objetivos 1 y 2). La ficha comenzó presentando tres ejemplos concretos sobre cómo realizar el diagrama de flujo para tres algoritmos: ordenación de dos números, cálculo de la suma, resta, multiplicación y división de dos números y por último, clasificación de un número en primo o compuesto. Dichos diagramas de flujo, fueron explicados por el profesor. A continuación se propuso que cada grupo realizara el diagrama de flujo correspondiente al algoritmo de Euclides. Para analizar los resultados de esta ficha, se diseñó una tabla de categorías (Tabla 2) que permitió clasificar las respuestas obtenidas.

La duración asignada a esta ficha fue una sesión de 55 minutos cada una: la primera para la ficha 3 se llevó a cabo en el aula de informática. El objetivo de la misma fue conseguir que los alumnos programaran en Scratch el algoritmo de Euclides para el cálculo del Máximo Común Divisor de dos números naturales. La ficha comenzó mostrando el código en Scratch de los tres algoritmos que se utilizaron en la ficha anterior para ejemplificar los diagramas de flujo. Los alumnos tenían que introducirlos en Scratch y validar su funcionamiento ejecutándolos varias veces. A continuación se les pidió que intentaran diseñar el código del algoritmo de Euclides. Se reservaron para esta ficha dos sesiones de 55 minutos cada una. La primera para la realización de los tres primeros ejercicios y la segunda para el desarrollo del código del algoritmo de Euclides. Para esta última, el profesor les repartió de nuevo su ficha 2 fotocopiada, ya que podían hacer uso del diagrama de flujo que diseñaron en la misma para implementar el programa en Scratch (se repartió la ficha fotocopiada ya que no nos interesaba que fuera modificada para su posterior análisis).

Tabla 2. Tabla de categorías para el análisis de la ficha 2

CATEGORÍAS	SUBCATEGORÍAS
C1. Identificación de las representaciones del diagrama de flujo	<p>C1.1. Los polígonos del diagrama de flujo no se corresponden en absoluto con la instrucción que deben representar.</p> <p>C1.2. Falta algún polígono o alguno de ellos no es el que debiese ser.</p> <p>C1.3. Todos los polígonos del diagrama de flujo se corresponden con el tipo de función que desempeñan (Inicio/Fin, Entrada/Salida de datos, Proceso o Decisión). No obstante, falta alguna línea de flujo en el diagrama.</p> <p>C1.4. Todos los polígonos del diagrama de flujo se corresponden con el tipo de función que desempeñan (Inicio/Fin, Entrada/Salida de datos, Proceso o Decisión). Además, aparecen todas las líneas de flujo.</p>
C2. Realmacenamiento las variables si $N1 < N2$	<p>C2.1. No realiza este paso en el diagrama de flujo.</p> <p>C2.2. Realmacena las variables, pero lo hace de forma incorrecta.</p> <p>C2.3. Realmacena las variables de forma correcta, pero no indica cómo se procederá en la programación</p> <p>C2.4. Realmacena las variables de forma correcta.</p>
C3. Identificación de la condición de parada	<p>C3.1. No aparece en el diagrama de flujo ninguna decisión que se corresponda con una condición de parada.</p> <p>C3.2. Identifica que existe un bucle, pero no establece ninguna condición de parada.</p> <p>C3.3. Identifica que existe un bucle y establece una decisión de parada, pero esta es incorrecta.</p> <p>C3.4. Identifica correctamente la existencia de un bucle y su condición de parada correspondiente.</p>
C4. Reasignación de las variables dentro del bucle	<p>C4.1. No existe reasignación de variables dentro del bucle.</p> <p>C4.2. Se reasignan variables dentro del bucle, pero de forma incorrecta.</p> <p>C4.3. Se reasignan variables dentro del bucle de forma correcta, pero el orden no es el adecuado</p> <p>C4.4. Se reasignan las variables correctamente dentro del bucle.</p>
C5. Devolución del MCD	<p>C5.1. No se muestra al usuario ninguna información de salida, por lo que el programa no devuelve correctamente el MCD.</p> <p>C5.2. Se muestra al usuario una información distinta del valor del último divisor, por lo que el programa no devuelve correctamente el MCD.</p> <p>C5.3. Se muestra al usuario el valor del último divisor, aunque el programa no devuelve correctamente el valor del MCD, al existir fallos a lo largo del algoritmo.</p> <p>C5.4. El programa devuelve correctamente el MCD.</p>

La ficha 4 supuso la institucionalización del diseño del algoritmo de Euclides a través del diagrama de flujo. Los alumnos debían comparar el diagrama de flujo que se les entregaba en esta ficha con el que realizaron en la ficha 2 para subsanar los posibles errores que habían cometido. El objetivo de esta ficha era intentar conseguir que todos los alumnos fueran capaces de implementar bien el programa en Scratch y tuvo una duración de 25 minutos.

La ficha 5 supuso la institucionalización del programa en Scratch ya que permitió calcular el MCD de dos números naturales. En dicha ficha se hizo entrega del código que permitía programar el algoritmo de Euclides. Los alumnos tenían que compararlo con el suyo y corregir los errores que tuvieran. A continuación se presentaron tres ejercicios en los que se pedía calcular el MCD de varios pares de números; además, también se solicitó que contestaran a cuestiones relativas sobre los resultados que se esperaban. El objetivo de esta ficha era que los alumnos fueran capaces de formular hipótesis que permitieran ser contrastadas a través del algoritmo diseñado e implementado (objetivo 3), además de introducir a los alumnos en el concepto de generalización a partir de la comprobación de casos particulares en la aplicación creada (objetivo 4). Esta ficha tuvo una duración de 30 minutos.

Para cuantificar los resultados obtenidos por cada grupo se diseñó el criterio de puntuaciones que figura en la tabla 3:

Tabla 3. Tabla de criterio de puntuaciones para el análisis de la ficha 4

PUNTACIÓN	CRITERIO
0	No devuelven correctamente el M.C.D para los pares de números que se muestran y, consecuentemente, no pueden formular correctamente hipótesis ni generalizar a partir de la comprobación de particulares.
2,5	Devuelven correctamente el M.C.D para los pares de números que se muestran pero no formulan correctamente hipótesis ni son capaces de generalizar a partir de la comprobación de particulares.
5	Devuelven correctamente el M.C.D para los pares de números que se muestran y son capaces de formular hipótesis de forma parcial o muy parcial.
7,5	Devuelven correctamente el M.C.D para los pares de números que se muestran y son capaces de formular hipótesis de forma correcta para dos de las tres preguntas que se proponen.
10	Devuelven correctamente el M.C.D para los pares de números que se muestran y son capaces de formular hipótesis de forma correcta, además de generalizar a partir de la comprobación de particulares.

La ficha 6 debía suponer una mejora de la aplicación creada en la ficha 5, ya que debía permitir calcular tanto el MCD como el MCM de dos números naturales. Los objetivos de la misma eran que los alumnos valorasen el concepto de modelo matemático (objetivo 5), además de hacerles formular hipótesis que fueran contrastadas a través del algoritmo diseñado e implementado (objetivo 3) e introducir a los alumnos en el concepto de generalización (objetivo 4). Se pedía que diseñaran el diagrama de flujo que permitía

calcular el MCM y también el código del programa que realizaría en Scratch el cálculo del MCM de dos números. La ficha acabó con varios ejercicios para que los alumnos probaran la aplicación creada y formularan hipótesis sobre los resultados que observaban. Los alumnos debían considerar el programa diseñado como un modelo matemático que permitía calcular el MCD y el MCM de dos números naturales. Se incidió en que dicho modelo podía ser mejorado. Esta ficha tuvo una duración de 55 minutos y se realizó en el aula de informática.

Nuestra metodología propone la introducción del Pensamiento Matemático Avanzado, ya que se manejan conceptos como el de algoritmo, diagrama de flujo, bucles de programación, etc, que requieren un grado de abstracción bastante importante. También hay una formalización matemática que queda expresada a través del código del programa que permite crear la aplicación y que supone una traducción de expresiones matemáticas del lenguaje usual al lenguaje algebraico. La formulación de hipótesis se realiza de manera continua y la comprobación de las mismas se realiza al final de cada ejercicio de programación y de diseño del diagrama de flujo. La generalización entra en juego cuando los alumnos comprueban que el programa funciona para los diferentes pares de números elegidos. Además de trabajar elementos propios del Pensamiento Matemático Avanzado, la forma de trabajar se adecua a la Teoría de las Situaciones Didácticas. Cada ficha es una situación didáctica en la que los alumnos tienen que trabajar por grupos. Durante estas fichas se producen los momentos de acción (los alumnos interaccionan con el problema a resolver), de formulación (dentro de cada grupo, cada alumno propone al otro su idea sobre cómo resolver el problema) y de validación (el alumno puede comprobar si el algoritmo es correcto o no, sin ayuda del profesor, simplemente ejecutándolo para varios casos distintos). La fase de institucionalización se produce cuando el profesor entrega el diagrama de flujo para el algoritmo de Euclides en la ficha 4 y el código del programa en la ficha 5. Todas estas situaciones didácticas forman parte de la situación didáctica que pretende la enseñanza del algoritmo de Euclides a través del software Scratch.

RESULTADOS

En este apartado se analizarán solamente los resultados relativos a la tercera ficha, con el fin de simplificar el análisis. Dicho análisis se realizará teniendo en cuenta las producciones de los alumnos sobre la programación del algoritmo de Euclides en Scratch.

Como se puede observar en la figura 1, el grupo 1 programó el algoritmo de Euclides prácticamente de forma correcta. Arreglaron la asignación de variables dentro del bucle, con respecto a lo que elaboraron en el diagrama de flujo e introdujeron de forma correcta la condición de parada. No obstante, como se puede observar, el programa no devuelve un resultado correcto cuando el dividendo es menor que el divisor, ya que no utilizan una variable auxiliar dentro del condicional para intercambiar las variables dividendo y divisor, error que también manifestaban en el diagrama de flujo y que no fue corregido en la programación. Penalizamos este error con 1 punto, obteniendo dicho grupo una puntuación de 9.

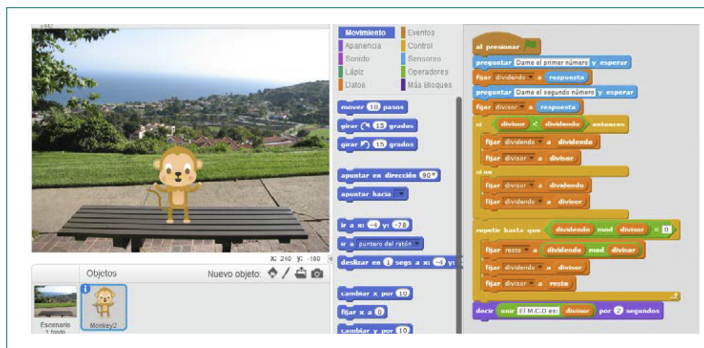


Figura 1. Resultados de la ficha 3 del grupo 1

El grupo 2 tuvo muchos más errores en la programación que el grupo anterior. Por un lado, la reasignación de variables dentro del condicional es incorrecta ya que asignan tanto a la variable dividendo como a la variable divisor, la variable dividendo. Por otro lado, la condición de parada del bucle se presenta de forma correcta, pero no asignan correctamente los valores a las variables dividendo y divisor. Penalizamos estos errores con 4 puntos (2 puntos cada uno), obteniendo dicho grupo una puntuación de 6.

Los alumnos del grupo 3, no asistieron a clase en la sexta sesión; por tanto, no se ha podido proceder al análisis de sus resultados.

El grupo 4 presentó un programa muy parecido al del grupo 1. En el caso en el que el dividendo es mayor que el divisor, el programa no funciona correctamente, ya que al no utilizar una variable auxiliar, asignan tanto a dividendo como a divisor el mismo valor en el condicional. Por otro lado, aunque el programa funciona para el caso en el que el dividendo es menor que el divisor, calculan el resto de dividir el número mayor entre el menor; algo poco intuitivo. Esto provoca una iteración más en el algoritmo y por tanto, una menor eficiencia de la que debiese tener. Penalizamos estos errores con 2 puntos (1 punto cada uno), obteniendo dicho grupo una puntuación de 8.

El grupo 5 comete el mismo error que el grupo 4 en la asignación de las variables dividendo y divisor dentro del condicional. Además, este grupo hace una asignación incorrecta de la variable divisor dentro del bucle (asignan divisor a dividendo), lo que provoca que el programa no devuelva un valor correcto para el MCD. No obstante, cabe mencionar que aunque el programa no funciona, introdujeron mejoras con respecto a lo que diseñaron en el diagrama de flujo. Penalizamos estos errores con 3 puntos (1 punto el primero y 2 puntos el segundo), obteniendo dicho grupo una puntuación de 7.

Los alumnos del grupo 6, no asistieron a clase en la sexta sesión; por lo tanto, no se ha podido proceder al análisis de sus resultados.

El grupo 7, al igual que sucedió en su diagrama de flujo, no introducen el condicional para situar en la variable dividendo el mayor de los números y en la variable divisor al menor. La condición de parada del bucle es correcta, pero calcularon dentro de éste el cociente en lugar de reasignar dividendo y divisor. Además, devuelven continuamente valores por pantalla, al situar una instrucción “decir” dentro del bucle. Los integrantes de este grupo no han asimilado que el programa únicamente debe devolver el último

La siguiente tabla resume los resultados obtenidos.

Tabla 4. Tabla de puntuaciones

GRUPO	PUNTUACIÓN
1	9
2	6
3	-
4	8
5	7
6	-
7	3
8	6
9	6
10	7
11	2
12	9

CONCLUSIONES Y PERSPECTIVAS FUTURAS

En este documento hemos trabajado el MCD y el MCM, conceptos que aparecen por primera vez en 5º de Primaria y que siguen trabajándose en los primeros cursos de Secundaria. Aunque por la forma habitual de trabajar estos conceptos en el aula formarían parte de lo que denominamos Pensamiento Matemático Elemental, la manera de trabajarlos que proponemos en este documento los englobaría dentro del Pensamiento Matemático Avanzado, ya que con nuestra metodología de trabajo se favorece el razonamiento lógico-matemático, la abstracción, la generalización, la formulación de hipótesis y la justificación de las mismas. Además, el diseño de las fichas de trabajo sigue los principios de la Teoría de Situaciones Didácticas, ya que, además de favorecerse el trabajo en grupo, durante las mismas se producen los momentos de acción, de formulación y de validación. La fase de institucionalización se produce cuando el profesor entrega el diagrama de flujo para el algoritmo de Euclides en la ficha 4 y el código del programa en la ficha 5. A continuación se detallan la consecución de los objetivos:

- Consideramos que el objetivo 1 se ha alcanzado de forma satisfactoria. La mayor parte de los alumnos comprendió el funcionamiento del algoritmo de Euclides, ya que la media de las puntuaciones de todos los grupos en la ficha 1 es 7,54. Además, fueron capaces de diseñar el diagrama de flujo del mismo en un alto porcentaje de los casos; esto es así ya que 4 de los grupos obtienen una puntuación

superior al 50% de la puntuación alcanzable y 3 obtienen una puntuación entre el 40% y el 50% de lo posible en el diseño del mismo.

- El objetivo 2 también consideramos que se ha cumplido. Un alto porcentaje de los alumnos, han sido capaces de diseñar el algoritmo de Euclides en Scratch (la media de las puntuaciones obtenidas por los grupos en el diseño del mismo es 6,3 (véase tabla 4). Además, tras el análisis de resultados de la ficha 3, observamos que, en general, los alumnos manifiestan un mejor resultado programando con Scratch que diseñando el diagrama de flujo. Intuimos que esto es por la validación que ofrece el lenguaje, que permite modificar el código hasta que se obtiene un resultado que se espera correcto. Por otro lado, también queremos señalar que ningún grupo utilizó la variable auxiliar para intercambiar las variables dividiendo y divisor en el condicional. Quizá la creación de dicha variable requiera un grado de abstracción que los alumnos de 3º ESO no han logrado todavía adquirir.
- Tras el análisis de las fichas 5 y 6, observamos que los alumnos tienen serias dificultades para formular hipótesis (objetivo 3) y para generalizar (objetivo 4). Esto puede ser debido a la excesiva mecanización de algoritmos que se realiza diariamente con los estudiantes de secundaria y al poco ejercicio de reflexión al que se les somete. Por tanto, podríamos decir que los objetivos 3 y 4 no se han alcanzado con el nivel que esperábamos. En concreto, encontramos un grupo que sí habría alcanzado estos objetivos con un nivel notable (grupo 2), cinco grupos que los habría alcanzado de forma suficiente (grupos 4, 5, 7, 9 y 12) y seis grupos que no habrían alcanzado los objetivos mencionados (grupos 1, 3, 6, 8, 10 y 11).
- Nuestro objetivo 5 se ha conseguido ya que todos los alumnos comprendieron que, en el caso en que el algoritmo estuviese bien programado, proporcionaría siempre el valor del MCD y MCM para cualquier par de números naturales introducidos por pantalla.

Para acabar, señalamos que un concepto formará parte del Pensamiento Matemático Elemental o Avanzado dependiendo del tratamiento que se dé al mismo. Edwards, Dubinsky y McDonald (2005) señalaban que si un concepto que por su naturaleza debiese formar parte del Pensamiento Matemático Avanzado, es tratado de manera procedimental, dicho concepto estaría dentro del Pensamiento Matemático Elemental. En este documento hemos tratado el caso contrario: un concepto que por su naturaleza estaría dentro del Pensamiento Matemático Elemental, puede formar parte del Pensamiento Matemático Avanzado si éste es tratado con una metodología que favorezca la aparición de elementos del mismo. En este caso, la programación es el instrumento que nos ayuda a trabajar el MCD desde este punto de vista.

Como perspectivas futuras se encuentra la puesta en práctica en 3º E.S.O de otra secuencia didáctica, distinta de esta, para calcular el MCD de dos números naturales, que favorezca también el desarrollo del Pensamiento Matemático Avanzado y complemente la anterior. La secuencia consistiría en el diseño en Scratch un programa que calcule los divisores de dos números pedidos por pantalla y devuelva el mayor de ellos. Creemos que dicha secuencia complementaria, podría favorecer la consecución de los objetivos 3 y 4, no alcanzados en ésta.

REFERENCIAS

- Adell, J. (1997). Tendencias en educación en la sociedad de las tecnologías de la información Edutec. Revista electrónica de tecnología educativa, (7).
- Brousseau, G. (1986). Fundamentos y métodos de la didáctica de las matemáticas *Recherches en didactique des mathématiques*, 7(2), 33-115.
- Brousseau, G. (1994). Los diferentes roles del maestro. En Parra, C. y Saiz, I. (Comp.), *Didáctica de Matemáticas. Aportes y reflexiones*. Buenos Aires: Paidós Educador.
- Brousseau, G. (1998). *Théorie des situations didactiques*, [Textes rassemblés et préparés par N. Balacheff, M. Cooper, R. Sutherland, V. Warfield], Grenoble: La Pensée Sauvage-Éditions, coll *Recherches en didactique des mathématiques*.
- Brousseau, G. (2011) *La théorie des situations didactiques en mathématiques* (Vol. 5, No. 1, pp. 101-104). Presses universitaires de Rennes.
- Carralero, N. (2011). Scratch. Programación fácil para primaria y secundaria. *Revista digital sociedad de la Información*, 29, 1-10
- Dreyfus, T. (1991). Advanced mathematical thinking processes En Tall, D. (Ed), *Advanced mathematical thinking*. Dordrecht: Kluwer Academic Publisher, 25-41.
- Edwards, B.S., Dubinsky, E., y McDonald, M.A. (2005). *Advanced mathematical thinking Mathematical Thinking and Learning*, 7(1), 15-25.
- Fernández, J., y Muñoz, J. (2007). Las TIC como herramienta educativa en matemáticas. *Unión*, 119-147.
- Gairín, J.M., y Sancho, J. (2002). *Números y algoritmos* Madrid: Síntesis.
- Ifrah, G. (2008). *Historia universal de las cifras*. Madrid: Espasa Calpe.
- Marmolejo, J.E., y Campos, V. (2012). Pensamiento lógico matemático con scratch en nivel básico *Vínculos*, 9 (1), 87-95.
- Panizza, M. (2003). *II Conceptos básicos de la Teoría de Situaciones Didácticas*. Disponible en Internet: http://crecerysonreir.org/docs/Matematicas_teorico.pdf
- Pascual, M.A. (1998). *La nueva frontera educativa con nuevas tecnologías. Nuevas Tecnologías, medios de comunicación y educación. Formación inicial y permanente del profesorado*. Madrid: CCS.
- Pérez, R.P. (1998). *Nuevas tecnologías y nuevos modelos de enseñanza. Nuevas tecnologías, medios de comunicación y educación: formación inicial y permanente del profesorado* (pp. 105-150). Madrid: CCS.
- Ricoy, M.C., y Couto, M.J. (2012). El acercamiento al contexto profesional como móvil para indagar sobre las TIC: un estudio cualitativo. *Revista Complutense de Educación*, 23 (2), 443-461.
- Robert, A. y Schwarzenberger, T. (1991). *Research in the teaching and Learning Mathematics at an Advanced Level*. En Tall, D (Ed.): *Advanced Mathematical thinking* (127-139). Springer Netherlands..
- Tall, D. (1985). *Understanding the calculus*. *Mathematics Teaching*, 49-53.
- Tall, D. (1991). *The psychology of advanced mathematical thinking*. En Tall, D. (Ed.): *Advanced mathematical thinking* (pp. 3-21). Springer Netherlands.
- Usiskin, Z. (1998). *Paper-and-Pencil Algorithms in a Calculator-and-Computer Age*. En L. J. Morrow y M. J. Kenney (Eds.) *The Teaching and Learning of Algorithms in School Mathematics* (pp. 7-20). Reston, VA: National Council of Teachers of Mathematics.