

# Estudio sobre diferencias de género en las competencias y las estrategias educativas para el desarrollo del pensamiento computacional

## A Study on Gender Differences in the Skills and Educational Strategies for the Development of Computational Thinking

Elisenda Eva Espino Espino

Instituto Universitario de Estudios de las Mujeres (IUEM). España.

alu0100940138@ull.edu.es

Carina Soledad González González

Instituto Universitario de Estudios de las Mujeres (IUEM). España.

cjgonza@ull.es

### Resumen

El pensamiento computacional es una competencia que debería ser incluida en la formación de todos los niños y niñas de las diferentes etapas educativas, desde las iniciales hasta las superiores. Sin embargo, cada vez son menos mujeres las que desarrollan esta capacidad porque no eligen en sus estudios superiores las carreras que se relacionan con la informática. Por ello, este artículo analiza las principales iniciativas de la enseñanza del pensamiento computacional, así como la brecha de género existente en la enseñanza de la informática. Para conocer las estrategias de enseñanza de pensamiento computacional y las diferencias de género halladas, se ha efectuado un estudio sobre las opiniones del profesorado, jueces, árbitros y voluntariado en la competición nacional de robótica educativa FLL (First Lego League) celebrada en Santa Cruz de Tenerife en 2015. Los resultados indican que, aunque hay mayor presencia masculina en el torneo, no hay diferencias significativas de género en las habilidades relacionadas con la programación y el aprendizaje del pensamiento computacional, pues chicos y chicas procesan y aprenden información equitativamente.

### Palabras Clave

Pensamiento Computacional, Enseñanza de la Programación, Brecha de Género, Estudio

### Abstract

Computational thinking is a competence that should be included in the training of every student of different educational stages, from the lower to the higher stages. However, there are fewer women that choose in their post-secondary education careers related to computer science. Therefore, this article analyzes the main initiatives of teaching the computational thinking, and also the gender gap in computer education. In order to know strategies of computational thinking and the gender differences which were found; it has been made a study about the opinions of teachers, judges, referees and volunteers of the national educational robotics competition FLL (First Lego League), whom was held in Santa Cruz de Tenerife in 2015. The results indicate that although there are more male presence in the tournament, there are no significant gender differences in the skills related to programming and learning computational thinking, because boys and girls learn and process information fairly.

### Keywords

Computational Thinking, Teaching of Programming, Gender Gap, Study

## Introducción

El ser humano está cada vez más interesado por la búsqueda del conocimiento mediante distintos métodos e incentivos educativos, convirtiéndose en un campo de estudio muy relevante en la actualidad. No sólo son los profesionales de la docencia los que se interesan por este tema, también lo son las familias ya que pretenden lograr una formación integral y contextualizada de sus hijos e hijas (M. de E. Nacional, 2006). Ellos y ellas son realmente los protagonistas principales del proceso de enseñanza-aprendizaje, tal y como expone Madrona (2003) y, por ello, surge el compromiso de atender a sus necesidades e intereses (La Casa, E., Lorente, E. 2003). Estos se verán estimulados por la posible afinidad o motivación hacia unos determinados contenidos o prácticas y un modelo de enseñanza que no se limita únicamente al conocimiento de la materia y/o la utilización de una u otra metodología, ni siquiera en saber el por qué y para qué enseñar, sino que también guarda estrecha relación con las emociones y los sentimientos (Devís, 1999).

En los últimos años, se ha evidenciado que la sociedad actual se encuentra inmersa en una fuerte era tecnológica, quedando los esquemas de análisis con los que observábamos el mundo obsoletos para comprender nuevas realidades (Tezanos, 2001). Así, no queda más remedio que la aceptación de ver el mundo desde nuevos parámetros tecnológicos y de ahí ha surgido una notoria proliferación de las Tecnologías de la Información y Comunicación (TIC) en los centros educativos de todo el mundo. Éstas están transformando notablemente tanto la forma de enseñar como la de aprender y, por supuesto, el rol del docente y el del estudiante (Gómez, L., Macedo, J. 2010); algo que obliga a pensar, no sólo en lo que se puede ofrecer al niño o niña para hacer un determinado procedimiento (software educativo), sino que debe orientarse hacia el desarrollo de ciertas capacidades para la vida como es el aprendizaje mismo del aprender, la adquisición de los mecanismos y formas de resolver los problemas, aplicando un proceso de abstracción concreto y práctico que facilite la toma de decisiones. A su vez, desarrollar habilidades sociales que faciliten el trabajo colaborativo y liderazgo en los estudiantes, es un objetivo fundamental en la educación de hoy en día (Hurtado, J.A.; Collazos, C.A.; Cruz, S.T.; Rojas, O.E., 2012), así como el aprender a resolver problemas computacionales de forma efectiva, pues estas son aplicables a multitud de contextos distintos tanto personal, social y/o académico (Feierherd, G.E.; Depetris, O.; Jerez, M., 2001).

De este modo, otra de las competencias que debería ser incluida en la formación de todos los niños y niñas desde las primeras etapas es la del Pensamiento Computacional, concepto totalmente ligado a las Tecnologías de la Información y Comunicación (TIC). Esto es así porque el pensamiento computacional posee una serie de características propias del pensamiento científico aplicadas al proceso de resolución de problemas mencionado con anterioridad. Dichas características son: permite la formulación y

resolución de problemas a través del uso de ordenadores y otras herramientas tecnológicas de soporte, favorece la organización, el análisis de la información de forma lógica, la representación de la información a través de abstracciones tales como modelos y simulaciones, la automatización de soluciones haciendo uso del pensamiento algorítmico (estableciendo una serie de pasos ordenados para llegar a la solución), la identificación, análisis e implementación de posibles soluciones con el objetivo de lograr la combinación más efectiva y eficiente de pasos y recursos y, por último, la generalización y la transferencia del proceso de resolución de problemas que permitan resolver diferentes familias de problemas. Si se explica de un modo más didáctico para facilitar su comprensión, se puede decir que el pensamiento computacional consiste en resolver un problema complejo usando otro que ya sabemos resolver, bien sea por reducción, por composición, por transformación o por simulación. También, se refiere a pensar recursivamente, es decir, interpretando códigos como información e información como códigos. A su vez, podría ser considerado también como la solución de un problema en términos de sí mismo hasta solucionar el trivial, más por su claridad y elegancia, que por su eficiencia. Además se puede conocer por el hecho de usar la abstracción y la descomposición cuando abordamos una tarea compleja o cuando diseñamos un sistema complejo y por escoger la representación adecuada para un caso determinado o modelar los aspectos relevantes de una problemática para hacerlo manejable (Zúñiga, R. & Hurtado, J., 2015). Igualmente, combinar la recursión y la abstracción ofrece una gran herramienta para construir soluciones por capas de abstracción, con lo que se facilita enfocar los esfuerzos en cada capa y posteriormente, pensar en las relaciones entre ellas hasta llegar a la solución deseada.

Una iniciativa muy interesante en relación a la definición del pensamiento computacional es la promovida por la Sociedad Internacional de la Tecnología en la Educación (ISTE) y la Asociación de Profesores de Informática (CSTA), que han colaborado con líderes del mundo de la investigación y la educación superior, la industria y la educación primaria y secundaria para desarrollar una definición operativa que describa con precisión sus características esenciales y ofrezca un marco de trabajo y un vocabulario común con el que los profesionales de la educación puedan trabajar (Moreno, 2014). Según dicha definición operativa, el pensamiento computacional es un proceso de resolución de problemas que incluye las siguientes características: formular problemas de forma que se permita el uso de un ordenador y otras herramientas para ayudar a resolverlos, organizar y analizar lógicamente la información, representar la información a través de abstracciones como los modelos y las simulaciones, automatizar soluciones haciendo uso del pensamiento algorítmico (estableciendo una serie de pasos ordenados para llegar a la solución), identificar, analizar e implementar posibles soluciones con el objetivo de lograr la combinación más efectiva y eficiente de pasos y recursos y, finalmente, generalizar y transferir este proceso de resolución de problemas para ser capaz de resolver una gran variedad de familias de problemas.

Debido a que estamos atravesando por esta novedosa etapa, las competencias relacionadas con la programación se están considerando destrezas básicas e instrumentales en la Sociedad del Conocimiento junto con las clásicas de lecto-escritura y matemáticas, algo que rompe con la cultura tradicional del proceso de enseñanza-aprendizaje y exige rápidas adaptaciones por parte del colectivo educativo ante el surgimiento de aceleradas modificaciones. De esta forma, podemos ver que diferentes países se sitúan como pioneros en la enseñanza de programación porque ya han comenzado a desarrollar acciones para promover la adquisición del pensamiento computacional, tales como Nueva Zelanda, Corea del Sur, los EE.UU, Israel y el Reino Unido. Sin embargo, todavía para el resto de países no son prioritarias ni se ha extendido su presencia en sus políticas educativas. A pesar de esto, existe algunos hitos recientes que muestran la preocupación de incorporar enseñanzas de esta índole en la población, debido a que se considera que son imprescindibles para el desarrollo de la nueva economía digital. Por ejemplo, 2014 ha sido declarado Año del Código (Year of Code) y también en 2014, en Reino Unido se lanzó una nueva iniciativa para capacitar a los docentes en programación y así aumentar la calidad de su enseñanza para el alumnado. Esto se debe a que existe una preocupación de que a pesar de la crisis económica actual, hay y habrán en el futuro millones de puestos de trabajo sin cubrir en el sector tecnológico porque la población no está preparada para desempeñar estas funciones.

Por otra parte, a menudo surgen dudas en los y las profesionales de la docencia por el hecho de que exista la posibilidad de que los niños y niñas sean o no capaces de contribuir a su propio desarrollo del pensamiento computacional, pudiendo así facilitar o dificultar su labor. Ante esta situación, se demuestra que aunque la mayoría de los chicos y chicas que utilizan computadores no son programadores, a muchos de ellos y ellas les encantaría programar su propio software y/o personalizar el que ya tienen, algo a lo que alude Pane (1998). Por lo general, desean realizar programas que son similares a los que utilizan habitualmente en su vivienda o en la escuela tales como juegos o aplicaciones educativas que son altamente interactivas y de riqueza gráfica. Aunque para los pequeños y las pequeñas las ideas sobre la programación no son muy simples, su aprendizaje se convierte en una herramienta poderosa para desarrollar sus habilidades lógico-matemáticas y sociales (Feierherd, G. E.; Depetris, O.; Jerez, M., 2001). Sin embargo, para este tipo de tareas las herramientas adecuadas para su edad no son suficientes y no están totalmente soportadas por metodologías que involucren a los niños y niñas en un verdadero desarrollo de software (Pane, 1998). De ahí que surja la necesidad de seguir progresando y profundizando en el desarrollo del pensamiento computacional y todo lo que ello conlleva, con el objetivo de obtener mejores resultados y aumentar la calidad de la enseñanza.

## La programación y el género femenino

El avance científico y tecnológico es uno de los principales desafíos que la comunidad mundial debe afrontar en este nuevo milenio como ya hemos comentado. Sin embargo, cuando se habla de mujeres, la reacción inmediata es la de indicar la poca presencia de éstas en su desarrollo. Esto es un hecho que preocupa y que no debe pasar desapercibido, ya que si profundizamos en los estudios de ciencias, ingeniería e informática que es lo que ahora nos ocupa, nos percatamos de la necesidad de que las niñas aprendan competencias tecnológicas, en concreto, la competencia de pensamiento computacional (Wing, 2006); pues ellas son también responsables de la generación y el consumo de la tecnología presente en la sociedad actual, teniendo un importante papel en el proceso de creación y supervisión de las mismas aunque en ocasiones no se le preste atención. Lo que sucede es que a lo largo de toda la historia y hasta la actualidad, ha existido una cultura androcéntrica (Facio, A. & Fries, L., 1999) que ha prevalecido en la conciencia individual y colectiva de muchos y muchas, repercutiendo en la invisibilidad de destacables contribuciones femeninas (Kulis, S.; Flavio, F.; Nagoshi, J., 2010) y abriendo una profunda brecha de género en este campo, Orengo (2014). Por ello cabe plantearse, en primer lugar, si las mujeres llegan a la educación superior en igualdad de condiciones que los hombres dada la distinta socialización que experimentan (Ortiz, 2007). En segundo lugar, hay que prestar atención a los estereotipos sexuales presentes en nuestras vidas, desde el momento en que nacemos hasta que llegamos a la etapa adulta como expone Sarrió (2002). Por último, atender a la discriminación jerárquica, según la cual, mujeres capaces y brillantes son mantenidas en los niveles inferiores en la escala política, en negocios o en la ciencia; o topan con un “techo de cristal” que no pueden traspasar en su propia profesión (Sarrió, 2002).

Un hecho importante a tener en cuenta ante la mencionada brecha de género existente en estudios técnicos que quizás podría frenarla o solventarla (Orengo, 2014); es la incorporación de ejemplos femeninos relevantes a la educación obligatoria (infantil, primaria y secundaria) que motive, atraiga y sirva de incentivo a las chicas a la hora de cursar y profundizar en este tipo de materias. Se destacan algunas mujeres importantes en el ámbito tecnológico como ejemplo, de entre muchas otras, siendo estas: Ada Lovelace, la primera programadora; Hedy Lamarr, precursora del Wifi y Bluetooth; Jude Milhon, creadora del ciberpunk; Evelyn Berezin, creadora de procesadores de texto; Lynn Conway, pionera en chips microelectrónicos; Frances E. Allen, pionera en la automatización de tareas paralelas; Grace Murray Hopper, desarrolladora del primer compilador y/o Top Secret Rosies, programadoras del primer computador ENIAC.

A su vez, son muchos los estudios nacionales e internacionales como el informe PISA 2014 (Informe PISA, 2014) o la encuesta de Competencias de la Población Adulta de la OCDE de 2012 (Informe OCDE, 2012), los que revelan el ascendente rendimiento que experimentan las niñas y jóvenes en materias como matemáticas, física o tecnología; siendo incluso mayor que el de sus compañeros varones, Waksman (2005). Por ello, en

el presente estudio se hace especial hincapié en la influencia del pensamiento computacional como un condicionante en la adquisición de conocimientos y habilidades en programación pues, cuando hablamos de pensamiento computacional, nos referimos a un concepto que implica organizar y representar información de forma lógica, resolver problemas y automatizar soluciones con ayuda del pensamiento algorítmico, diseñar y aplicar herramientas y técnicas de la informática para comprender y razonar sobre los sistemas y procesos tanto naturales como artificiales que acontecen en el mundo que nos rodea, además de permitir el hecho comprender el comportamiento humano haciendo uso de los conceptos fundamentales de la informática, tal y como expone Wing (2006). Por lo tanto, fomentar en el género femenino el interés por la adquisición de esta competencia podría ser realmente útil a largo plazo para un equilibrado y equitativo desarrollo de la tecnología.

### **Estrategias, herramientas e iniciativas educativas para el desarrollo del pensamiento computacional**

Existen distintas iniciativas y herramientas educativas para enseñar el pensamiento computacional que se están llevando a cabo a nivel internacional, nacional y local. Casi todas tienen como punto en común la robótica educativa y la programación de videojuegos. Por ejemplo, presentamos a ChildProgramming (Cruz, T. R.; Hurtado, O.J.; Collazos C., 2013), un proyecto que brinda un marco para la enseñanza de la programación a través del fomento del trabajo en equipo siguiendo un enfoque basado en problemas. Aunque en ella se motiva el trabajo en equipo y la descomposición de tareas para alcanzar el objetivo final, el trabajo no estudia ni define estrategias computacionales (por ejemplo un método) para abstraer y descomponer la aplicación, así como para su posterior integración. En este tipo de estrategias se aplican metodologías ágiles de desarrollo de software como puede ser el entorno Scratch que se define como un entorno de aprendizaje de lenguaje de programación que permite a los principiantes aprender a escribir de manera sintácticamente correcta primero con la posibilidad de crear historias, juegos y simulaciones interactivas. A continuación da pie a la investigación, permite introducir y jugar con la programación de ordenadores utilizando una interfaz gráfica muy sencilla. Scratch ofrece un contexto y un conjunto de oportunidades para contribuir en la conversación activa sobre Pensamiento Computacional, pues las actividades de aprendizaje basadas en diseño (en particular, la programación de medios interactivos), apoya al desarrollo de este pensamiento en los y las jóvenes. Ese interés emergente hunde sus raíces en el compromiso de aprender mediante el diseño de actividades, enfoque constructivista del aprendizaje que resalta la importancia de que los niños y niñas se comprometan o involucren con el desarrollo de artefactos externos (Kafai & Resnick, 1996).

También existen otras herramientas como pueden ser Alice, un entorno gráfico para la iniciación a la programación utilizando ambientes 3D. Además, tenemos al alcance

Lego Mindstorms, juego de robótica para niños fabricado por la empresa de juguetes LEGO, el cual posee elementos básicos de las teorías robóticas, como la unión de piezas y la programación de acciones, en forma interactiva. Otro programa es Kids Ruby, que permite aprender programación mediante Ruby y puede considerarse como una evolución de Hackety Hack, el cual permite que los pequeños/as aprendan conceptos básicos de programación mediante un entorno gráfico muy atractivo. Otro programa muy útil puede ser Kodu, basado en un proyecto iniciado por Microsoft, permite crear juegos de manera visual desde el PC y la videoconsola Xbox. Ha sido diseñado para enseñar a programar basándose en tres conceptos básicos: creatividad, resolución de problemas y narrativa. Esta es solo una pequeña elección entre multitud de recursos más a los que se puede acudir a la hora de mejorar la enseñanza y el aprendizaje de la informática en el estudiantado. Para esto, Brennan & Resnick (2012) sugieren considerar las diferentes formas de conocimiento de los aprendices, ya que es realmente el aprendiz el que puede darle sentido al concepto cuando lo lleva a la práctica. Éstos autores sugieren que las evaluaciones del pensamiento computacional deben explorar las distintas formas de conocimiento.

Haciendo referencia a las estrategias educativas para el desarrollo del pensamiento computacional encontramos: ser incremental e iterativo (diseñar un proyecto no es un proceso limpio y secuencial en el que primero se identifica un concepto para el proyecto, luego se desarrolla un plan de diseño y posteriormente, se implementa el diseño del código. Es un proceso adaptativo, uno en el que el plan puede cambiar para responder a un enfoque de solución por pequeños casos), ensayar y depurar (rara vez las cosas funcionan como se había imaginado, por lo que es crítico para los diseñadores desarrollar estrategias para manejar y anticipar problemas), reusar y remezclar (construir sobre lo que otros ya han hecho, ha sido una práctica de vieja data en programación, que se ha amplificado con las tecnologías de redes que permiten acceder a un amplio rango de trabajos hechos por otros que se pueden reusar y remezclar) y, por último, abstraer y modularizar (se caracteriza por el hecho de construir algo grande uniendo colecciones de partes más pequeñas, es una práctica siempre importante para el diseño y la solución de problemas).

De las numerosas iniciativas existentes para la enseñanza de la programación en la infancia, destacaremos el torneo Firt Lego League (FLL), el cual constituirá nuestro caso de estudio. La FLL es un torneo que desafía a niños, niñas y a jóvenes de 10 a 16 años con una temática del mundo real relacionada con la ciencia y la tecnología. Cada año FLL propone un nuevo desafío, que tiene tres partes: el proyecto científico, el diseño y programación de un robot y los valores FLL (inclusión, descubrimiento, cooperación, respeto y trabajo en equipo). Para participar en el desafío los equipos se inscriben y organizan sus torneos microFLL, que dan acceso a los Torneos Clasificatorios FLL, realizándose competiciones en diferentes niveles: dentro del propio colegio, en la localidad, en la región, a nivel nacional y a nivel internacional. En este trabajo, nos planteamos estudiar las características y estrategias seguidas por los

equipos que habían sido capaces de superar los diferentes torneos hasta llegar al torneo nacional. A continuación, presentamos como hemos llevado a cabo dicho estudio.

## **Estudio**

Este trabajo de investigación tiene como objetivo analizar si existen diferencias de género en la enseñanza-aprendizaje de la programación, algo a lo que alude Justina, (2014). Para ello, necesitamos conocer cómo se está enseñando esta competencia en los colegios de las distintas etapas: educación infantil, educación primaria y educación secundaria; además de averiguar qué estrategias se llevan a cabo y que conceptos se trabajan en la enseñanza de programación. Este artículo se organiza de la siguiente forma: primero analizaremos las principales iniciativas de enseñanza del pensamiento computacional y la brecha de género existente en este tema. Luego, presentaremos un estudio llevado a cabo en una competición nacional de robótica educativa junto a sus principales resultados y las conclusiones extraídas.

Para poder conocer las estrategias llevadas a cabo para la enseñanza del pensamiento computacional y las diferencias de género, hemos analizado las opiniones del profesorado de los grupos de estudiantes ganadores del campeonato nacional (España) de la feria de robótica educativa FLL, llevada a cabo en Tenerife en marzo de 2015 (Figuras 1 y 2). Asimismo, se estudiaron las opiniones de los árbitros, del equipo de voluntariado y jueces que participaban en dicha competición, incluyendo como una de las variables a analizar, la introducción o no de la programación en las distintas etapas que abarcan desde la educación infantil hasta la educación secundaria obligatoria como una asignatura más del currículo educativo. Finalmente, se realizó una discusión de los resultados obtenidos y se presentaron las conclusiones del mencionado estudio.

De este modo, exploramos dos hipótesis principales, correspondiéndose con:

- Existen diferencias de género en el aprendizaje del pensamiento computacional siendo mayor la presencia de varones. ( $H_1$ )
- En el procesamiento y el aprendizaje de la información, no hay una diferenciación relevante que justifique que varían según el género. ( $H_2$ )



Figura 1. Grupo de niños y niñas defendiendo su proyecto científico y su robot ante el tribunal de jueces



Figura 2. Asistentes al Evento First Lego League en Santa Cruz de Tenerife.

## Método

### *Diseño*

Se ha realizado un estudio de tipo cuantitativo, selectivo y co-relacional por muestreo probabilístico aleatorio estratificado con conteo y determinación de proporciones.

### *Participantes*

La población encuestada se compone de 53 sujetos: 39 docentes, 5 árbitros y 9 personas dedicadas al voluntariado en eventos sobre programación educativa. Aludiendo al género (Tabla 1), las distintas agrupaciones que encontramos son: género de formadores, género de árbitros y género de voluntariado; denotándose en todos los agrupamientos una mayor asistencia masculina a excepción del voluntariado, donde existe menor diferencia destacando la presencia del género femenino.

	<b>Formadores</b>	<b>Árbitros</b>	<b>Voluntariado</b>
<b>Masculino</b>	56%	80%	40%
<b>Femenino</b>	44%	20%	60%

Tabla 1. Género de participantes

A su vez, el grupo de Formadores, el de árbitros y el de voluntariado, provienen de las siguientes Comunidades Autónomas, destacando por su mayoría, a Canarias y Catalunya (Tabla 2):

	<b>Formadores</b>	<b>Árbitros</b>	<b>Voluntariado</b>
<b>Andalucía</b>	3	-	-
<b>Canarias</b>	10	4	6
<b>Catalunya</b>	16	-	2
<b>Castilla y León</b>	5	-	-
<b>Navarra</b>	1	-	-
<b>País Vasco</b>	4	-	1

Tabla 2. Comunidades Autónomas

En formación adquirida, la mayor parte de los sujetos pertenecen a ramas técnicas. El 41% de los *docentes* son de Ingenierías Superiores; seguido de Ingenierías técnicas con un 20%, quedando detrás estudios de: magisterio, otras diplomaturas, licenciatura en matemáticas, física; licenciatura en psicopedagogía, pedagogía, arquitectura o doctorado. En el caso de los *árbitros*, se sitúan con un 80% en Ingeniería Técnica seguido de Ingenierías Superiores. Por último, el *voluntariado* se aleja de estos datos, caracterizándose por sumar un 66% de estudiantes y un 34% de diplomados en áreas no especializadas en contextos tecnológicos. En relación al grupo *docente*, es el que acumula mayor experiencia profesional. Se sitúa con un 69% en más de 5 años. Sin embargo, los *árbitros* y *voluntariado* no presentan años de docencia, salvo un 2% del primer grupo con menos de 1 año. A su vez, las personas con más titulaciones afines al campo técnico y tecnológico son los docentes con un 72%, seguidos por los árbitros que presentan un 60%. Haciendo referencia al voluntariado, pertenecen a otras áreas en un 90%.

<b>Especialización</b>	<b>Formadores</b>	<b>Árbitros</b>	<b>Voluntariado</b>
<b>SÍ</b>	72%	60%	10%
<b>NO</b>	28%	20%	90%

Tabla 3. Área de especialización por agrupamiento

### *Instrumentos*

Se confeccionaron tres cuestionarios (uno para el profesorado, otro para los árbitros y el último, para el voluntariado) con 6 apartados. Uno se dedicó a los datos referenciales (género, edad, residencia, formación, área de especialización y docencia). Otro, a la composición de los grupos de alumnado en el evento. El tercero, a los conceptos de programación que se trabajan con el alumnado. El siguiente, a las estrategias y metodología de programación adoptadas. Posteriormente, se aludió a las competencias trabajadas en el proceso de enseñanza-aprendizaje con esta actividad. A continuación, se introdujo una pregunta de respuesta múltiple para conocer opiniones acerca de la inserción de la asignatura de programación en el currículo educativo. Y un último apartado con una pregunta cerrada y otra abierta dedicado a valorar las posibles diferencias de género en el desarrollo del pensamiento computacional. Además, se efectuaron entrevistas informales con los tres grupos citados: formadores, árbitros y voluntariado y se realizó una observación procesual no estructurada del comportamiento de los alumnos y alumnas.

### *Procedimiento*

Los cuestionarios, junto a una explicación del estudio y una invitación a participar en el mismo, fueron entregados a los entrenadores responsables de los equipos participantes de la FLL el día del registro (día anterior al comienzo del evento). Asimismo, durante el desarrollo del evento se puso también a disposición de los formadores/as, árbitros y voluntarios/as. Una vez cumplimentados, se recogieron y se registraron telemáticamente en una base de datos que, posteriormente, se sometió a los pertinentes análisis estadísticos.

En relación a las entrevistas informales, se realizó un registro de las mismas y con respecto a la observación no estructurada, se tomaron anotaciones del comportamiento del alumnado durante el desarrollo de la liga FLL Nacional 2015.

### **Resultados**

Los primeros datos indican que la *composición del alumnado* de los equipos participantes es, mayoritariamente, masculina (64%).

Haciendo referencia al segundo apartado, los *conceptos de programación* se corresponden con: secuencias (son un concepto clave en programación, es que una tarea o actividad particular se expresa como una serie de pasos o de instrucciones individuales que puede ejecutar el computador), ciclos (son mecanismos que ejecutan la misma secuencia múltiples veces), eventos (es un component esencial de los medios interactivos y que desencadena que otra suceda), paralelismo (los lenguajes de computador más modernos soportan estos paralelismos, esto es, secuencias de

instrucciones que se suceden simultáneamente), condicionales (la habilidad de tomar decisiones con base en ciertas condiciones), operadores (ofrecen apoyo a las expresiones matemáticas, lógicas y de cadenas de caracteres, permitiendo al programador realizar manipulaciones numéricas y de cadenas), datos (incluyen guardar, recuperar y actualizar valores) y, por último, los bucles (es una sentencia que se realiza repetidas veces a un trozo aislado de código hasta que la condición asignada a dicho bucle deje de cumplirse). En este estudio se ha obtenido como resultado que los que más se suelen trabajar con los alumnos/as, por orden descendente son: los bucles, los condicionales y las secuencias. Podemos comprobar estos datos en la Tabla 4 del presente artículo en la que  $n_i$  son valores absolutos y  $f_i$  frecuencias relativas. Además, en este apartado se alude al trabajo de distintos lenguajes de programación y a la comunicación entre dispositivos.

Concepto	$n_i$	$f_i$
Secuencias	11	73%
Ciclos	8	54%
Paralelismo	2	13%
Eventos	6	40%
Condicionales	11	73%
Operadores	9	60%
Datos	6	40%
Bucles	12	80%
Otros	6	40%

Tabla 4. Conceptos de programación

Con respecto a las *estrategias para la programación* que aparecen en la Tabla 6, destacan: ensayar y depurar y la enseñanza de sensores, seguidos de ser incremental e iterativo y la programación lineal en el diseño del robot. Si se aluden a las estrategias para la programación del robot destacan los sensores (dispositivo diseñado para recibir información de una magnitud del exterior y transformarla en otra magnitud, normalmente eléctrica, que seamos capaces de cuantificar y manipular), bloques (modo de agrupar diferentes secciones de un programa), programación lineal (procedimiento o algoritmo matemático mediante el cual se resuelve un problema indeterminado, formulado a través de un sistema de inecuaciones lineales, optimizando la función objetivo) y programación por bloques (permite crear programas que controlan y mezclan imágenes, animaciones, sonidos e interactuar con el usuario)

Estrategia	$n_i$	$f_i$
------------	-------	-------

Ser incremental e iterativo	9	60%
Ensayar y depurar	14	93%
Reusar y re-mezclar	5	36%
Abstraer y modularizar	2	13%
Bloques	7	47%
Sensores	10	67%
Programación lineal	9	60%
Programación por bloques	7	44%

Tabla 6. Estrategias de programación

Profundizando en la *metodología* adoptada por los responsables de la docencia, se menciona la dedicación de un trimestre a la enseñanza de los conceptos básicos y, posteriormente, se acogen a un sistema totalmente práctico con creación de Apps, Robótica, etc. También, este grupo destaca la importancia del trabajo por proyectos y del trabajo cooperativo.

En relación a las *competencias que pueden favorecer el proceso de enseñanza-aprendizaje de la programación* comprobamos en la Tabla 7 las que más destacan: autonomía e iniciativa personal, aprender a aprender y el tratamiento de la información, competencia digital e innovación. Hay que aludir también a la importancia que los encuestados/as dan al trabajo en equipo y la educación en valores.

Competencia	n <sub>i</sub>	f <sub>i</sub>
Comunicación Lingüística	11	69%
Matemáticas	10	63%
Conocimiento e Interacción con el mundo físico	12	75%
Tratamiento de la información, competencia digital e innovación	13	81%
Social y ciudadana	10	63%
Cultural y artística	8	50%
Aprender a aprender	14	88%
Autonomía e iniciativa personal	15	94%

Tabla 7. Competencias en programación

En cuanto a la *inserción de la programación en el currículo educativo*, podemos ver reflejado en la Tabla 8 como los/las docentes en un 70% coinciden en que la materia se convierta en obligatoria, al igual que los árbitros con un 60%. El voluntariado, por el contrario, opina en un 89% que sería más conveniente que la asignatura sea optativa, sobretodo, por gustos y preferencias y/o capacidades individuales.

Inserción	Formadores		Árbitros		Voluntariado	
	n <sub>i</sub>	f <sub>i</sub>	n <sub>i</sub>	f <sub>i</sub>	n <sub>i</sub>	f <sub>i</sub>
Obligatoria	27	70%	3	60%	1	12%
No obligatoria	-	-	-	-	-	-
Optativa	11	28%	2	40%	8	89%
Extraescolar	2	5%	-	-	-	-

Tabla 8. Inserción de la programación en el Currículo Educativo

Por último, las tres agrupaciones (formadores, árbitros y voluntariado) indicadas en la Tabla 9 coinciden en que existen *diferencias de género en programación*. Aunque la mayoría de los formadores está de acuerdo con esta afirmación, es el grupo que mas fluctuaciones presenta.

Diferencias de Género	Formadores		Árbitros		Voluntariado	
	n <sub>i</sub>	f <sub>i</sub>	n <sub>i</sub>	f <sub>i</sub>	n <sub>i</sub>	f <sub>i</sub>
<b>SÍ</b>	25	64%	5	100%	9	100%
<b>NO</b>	14	36%	-	-	-	-

Tabla 9. Diferencias de Género en Programación

A su vez, se realiza un desglose por parte de las tres agrupaciones en relación a los puntos fuertes y débiles según el género del alumnado, masculino o femenino. Punto fuerte se entiende como aquella cualidad y/o habilidad en las que se destaca positivamente y que favorece el aprendizaje y desarrollo del pensamiento computacional. Punto débil se entiende como aquella cualidad y/o habilidad de la que se carece y que dificulta el aprendizaje y desarrollo del pensamiento computacional. En la Tabla 10 se observan los resultados del grupo de formadores, del grupo de los árbitros y del voluntariado.

	Punto Fuerte	Punto Débil
	<i>Formadores</i>	
<b>Masculino</b>	+ Rapidez y Motivación +Prácticos +Competitividad +Liderazgo	- Constancia - Organización - Orden - Trabajo en Equipo - Focalización de intereses
<b>Femenino</b>	+Atención +Constancia +Creatividad +Organización +Trabajo en Equipo	- Poca práctica - Liderazgo - Gustos
	<i>Árbitros</i>	

<b>Masculino</b>	+Participación	- Organización
<b>Femenino</b>	+Liderazgo +Organización	- Participación
<i>Voluntariado</i>		
<b>Masculino</b>	+Prácticos	- Organización
<b>Femenino</b>	+Trabajo en Equipo +Organización	- Carácter

Tabla 10. Diferencias de Género formadores, árbitros y voluntariado

Haciendo referencia a las aportaciones del grupo de formadores de este estudio, manifiestan que los chicos destacan por asumir roles más prácticos y las chicas por la adopción de roles más organizativos. En relación a sus deficiencias en el aprendizaje y desarrollo de la competencia de pensamiento computacional, ocurre lo contrario pues donde más destaca el género masculino es donde menos destaca el género femenino.

En el caso de los árbitros de este estudio, consideran que los chicos son más participativos que las chicas, siendo las últimas más organizadas y caracterizándose por asumir roles de liderazgo. Atendiendo a sus deficiencias ocurre exactamente lo mismo que en el agrupamiento anterior.

Aludiendo al voluntariado, destaca como valor que favorece la adquisición del aprendizaje y desarrollo del pensamiento computacional el trabajo en equipo y la organización en las chicas, mientras que en los chicos su predisposición hacia la práctica. Si se atiende a las deficiencias, los chicos no se caracterizan por su organización como coinciden los grupos anteriores y a las chicas les falta carácter a la hora de dar los primeros pasos en la defensa de un proyecto o asumir el mando en un equipo de trabajo, quizás ocasionado por la influencia de factores externos en su capacidad en la toma de decisiones.

## Conclusiones

En primer lugar, los resultados expuestos confirman la existencia de una mayor cantidad de asistentes a la liga First Lego League 2015 de género masculino que de femenino, algo que denota que aún en la actualidad el interés por aspectos relacionados con el aprendizaje de la programación es mayor en los hombres que en las mujeres, aunque esta situación está comenzando a cambiar. Sin embargo, no hay diferencias significativas en las habilidades relacionadas con la programación y el aprendizaje del pensamiento computacional. Si profundizamos en la existencia de diferencias de género en cualidades y/o habilidades relacionadas con la adquisición y el aprendizaje del pensamiento computacional, comprobamos que los puntos fuertes de los chicos son, mayoritariamente, los débiles de las chicas, y con los puntos débiles ocurre algo similar.

Si se analiza el género masculino, hay coincidencias entre las aportaciones de los/las docentes y el voluntariado en el punto fuerte que alude a que los chicos son más prácticos para realizar las tareas destacando los árbitros, además, su participación; mientras que si se alude al punto débil donde todos/as coinciden, destaca la organización. Por el contrario, estudiando el género femenino, el punto fuerte donde todos los grupos coinciden es la organización. En el caso negativo, se detecta en ellas una falta de participación, dominio y liderazgo. Todo ello unido a las anotaciones recopiladas mediante las entrevistas efectuadas y la observación etnográfica, descubrimos que los chicos suelen destacar en aspectos relacionados con la práctica y taller, la competitividad y el liderazgo pero fallan en la organización y el trabajo en equipo. Por el contrario, las chicas suelen ser buenas organizadoras, saben trabajar en equipo, son constantes y creativas pero desciende su interés en cuestiones relacionadas con el liderazgo. Si reflexionamos acerca de este asunto y esta clara inversión, planteamos que la creación de grupos mixtos podría ser beneficiosa porque habría una notable complementariedad de cualidades entre ellos y ellas, algo que podría aumentar el rendimiento de dichas agrupaciones. A su vez, tenemos presente que todos los resultados pueden verse modificados por la existencia de variables que influyen considerablemente como son: las diferencias individuales, las inteligencias múltiples, los estilos de aprendizaje, entre otros algo a lo que alude Ortiz (2007). Es importante comentar también que la adopción de los roles mencionados con anterioridad por parte de los distintos géneros podría estar influenciada por factores externos y estereotipos sociales (Sarrió, 2002), algo que habría que trabajar para poder lograr al fin una distribución más equitativa e igualitaria.

En segundo lugar, aludiendo al procesamiento de la información y su aprendizaje, no se ha encontrado diferencias notorias que justifiquen que varía según el género. Asimismo, mediante el estudio de las respuestas de los diferentes agentes estudiados se ha manifestado que en su práctica profesional no han observado diferencias entre alumnos y alumnas, sin embargo, esto puede ser debido a que no se ha estudiado directamente este fenómeno ni se ha prestado atención en profundidad. Por ello, planteamos la necesidad de continuar estudiando e investigando sobre este asunto y plantear otro tipo de estudios que ahonden en este fenómeno, con iniciativas, herramientas e instrumentos subjetivos como los que hemos utilizado en este estudio, así como otros instrumentos objetivos (por ejemplo, *eye tracking*, *brain computing*, etc.).

De este modo, el hecho de incluir el concepto de pensamiento computacional en las escuelas de los distintos países del mundo es cada vez más necesario y suena cada vez más cercano (Zúñiga, R. & Hurtado, J., 2015), aunque aún queda mucho por avanzar y lograr. En términos generales hemos recopilado las opiniones de la muestra analizada, observándose que el grupo de docentes y árbitros, concedores muchos y muchas en profundidad del área técnica, consideran que la asignatura de programación debería estar implícita en el currículo educativo según los datos obtenidos en la variable que alude a la inserción de la programación, en el apartado de resultados de este artículo.

Sin embargo, aquellos/as no especializados en este ámbito, como es el voluntariado, exponen que debe incluirse pero de forma optativa. Las razones dadas por parte de este último grupo, no expertos en tecnología, tiene que ver con su complejidad y con que los intereses del alumnado pueden no estar orientados a esta temática. Tras un proceso de entrevista con estos sujetos, se denotó que relacionan la programación con conceptos puramente matemáticos y abstractos. Sin embargo, los docentes y árbitros, expertos en tecnología, abordan la programación como una materia con gran contenido transversal y multitud de posibilidades y beneficios con respecto al resto de asignaturas (Informe OCDE, 2012). Respecto a la cantidad de alumnado en este tipo de competiciones sobre robótica y programación, encontramos una mayor presencia de sujetos masculinos que ascienden a un 64% del total según los datos de este estudio (Ortiz, 2007). Por ello, consideramos que aún hay que trabajar profundamente en el fomento de vocaciones científicas y de ingeniería en las niñas. En el transcurso del estudio, observamos que en las etapas de infantil y primaria existe mayor homogeneidad entre niños y niñas, mientras que a medida que aumenta el nivel educativo disminuye, quizás, por posibles intereses o estereotipos implantados.

En tercer lugar, con respecto a las estrategias utilizadas para el desarrollo del pensamiento computacional las que más destacan son la de ensayar y depurar, ya que es imprescindible en programación desarrollar diversas estrategias para poder manejar y solventar posibles problemas que puedan surgir y ser incremental e iterativo, pues el hecho de diseñar no es un proceso único, sino adaptativo respondiendo a una solución a través de la acumulación de pequeños pasos. Haciendo referencia exclusivamente a la programación del robot, son los sensores y la programación lineal quienes gozan de mayor protagonismo.

A su vez, en nuestras conversaciones con el grupo de formadores, árbitros y voluntariado escuchamos que observan una relación entre la enseñanza-aprendizaje de la programación con el hecho de la evolución en la auto-comprensión de los alumnos y alumnas, en su relación con los tics y respecto al mundo tecnológico que les rodea. Esta fue una dimensión sorprendente y fascinante, una dimensión que no se había contemplado en nuestra formulación de conceptos pero que merece ser destacada. De manera que como paso final en la articulación del Pensamiento Computacional, añadimos algunas perspectivas que podrían repercutir en los cambios de apreciación que se denotan en los y las jóvenes que trabajan programación pero del que no se ha elaborado un estudio exhaustivo en este caso. Algunas de ellas son la expresión (las personas están rodeadas de medios interactivos, pero la mayor parte de sus experiencias con estos es siempre como consumidores. Se invierte tiempo en señalar, buscar y enviar mensajes con otros individuos, actividades que se pueden considerar importantes para aprender a usar las TIC, pero no son suficientes para desarrollarnos como pensadores computacionales. Un auténtico pensador computacional ve la computación como algo más que un producto de consumo, es algo que pueden usar para diseñar y auto expresarse), la conexión (la creatividad y el aprendizaje son prácticas profundamente

sociales por lo que no sorprende que diseñar medios computacionales se enriquezca mucho mediante la interacción con los otros, ya sea cara a cara o mediante la comunidad en línea, algo a lo que aluden Brennan & Resnick (2010) en prensa) y la pregunta (con la perspectiva computacional de preguntar, buscamos indicadores de que los y las jóvenes no sientan esa desconexión entre las tecnologías que les rodean y sus habilidades para negociar las realidades del mundo tecnológico. Todo lo contrario, ellos y ellas deben sentirse empoderador para formular preguntas sobre y con la tecnología).

Por último y en relación a esta investigación, nos ha abierto puertas para poder seguir trabajando en aspectos relacionados con la temática de enseñanza de programación desde una perspectiva de género, como puede ser la relación de competencias que se podrían desarrollar con el fin de convertirlas en fortalezas para el proceso de enseñanza-aprendizaje, previniendo problemática estereotipada y mejorando resultados. A su vez, consideramos que aporta datos susceptibles de seguir profundizando ya que su abordaje es justificable en otros contextos educativos, siendo solo un eslabón más de una larga cadena con gran proyección de futuro. De este modo, deseamos que este estudio preliminar pueda servir de base para posteriores investigaciones sobre la enseñanza del pensamiento computacional con perspectiva de género.

Fin de redacción del artículo: 29 de julio de 2015

Fecha de aprobación del artículo: 28 de agosto de 2015

Fecha de publicación del artículo: 15 de septiembre de 2015

Espino, E. y González, C. (2015). Estudio sobre diferencias de género en las competencias y las estrategias educativas para el desarrollo del pensamiento computacional. *RED. Revista de Educación a Distancia. Número 46*. 15 de Septiembre de 2015. Consultado el (dd/mm/aa) en <http://www.um.es/ead/red/46>

## Referencias

- Brennan, K., Resnick, M. (2012). *New frameworks for studying and assesing the development of computational thinking*. Paper presented at annual American Educational Research Association meeting, Vancouver, BC, Canada.
- Brennan, K., Resnick, M. (2010). Imagining, creating, playing, sharing, reflecting: How online community supports Young people as designers of interactive media. In N. Lavigne & C. Mouza (Eds), *Emerging technologies for the classroom: A learning sciences perspective*. Springer
- Devís, J. (1999). Emoción, educación e innovación: la revision de un estudio de casos. *I Congreso Internacional de educación física*. Fondo Internacional de Enseñanza
- Facio, A., Fries, L. (1999). Feminismo, género y patriarcado. *Bibliovirtual*: <http://cidem-ac.org>

- Feierherd, G. E.; Depetris, O.; Jerez, M. (2001). Una evaluación sobre la incorporación temprana de algorítmica y programación en el ingreso a Informática. *VII Congreso Argentino de Ciencias de la Computación*.
- Gómez, L., Macedo, J. (2010). Importancia de las TIC en la educación básica regular. *Tecnología de la información* 12 (25), 209-224. ISSN 1728-5852
- Hurtado, J. A.; Collazos, C. A.; Cruz, S. T.; Rojas, O. E. (2012). Child Programming: una estrategia de aprendizaje y construcción de Software basada en la lúdica, la colaboración y la agilidad. *Revista Universitaria RUDIC*, 1(1)
- Informes PISA (2014). Accesible en: <http://bit.ly/1cXFBAh>
- Informe OCDE (2012). Accesible en: <http://bit.ly/1C38VBA>
- Justina, H. (2014). Gender differences in Scratch Game design. *3rd International Conference on Information, Business and Education Technology (ICIBET 2014)*, 100-103
- Kafai, Y.B., Resnick, M. (1996). Constructionism in practice: Designing, thinking and learning in a digital world. Hillsdale, NJ: Erlbaum
- Kulis, S., Flavio, F., Nagoshi, J. (2010). Gender Roles, Externalizing Behaviors, and Substance Use Among Mexican-American Adolescents. *Journal of Social Work Practice in the Addictions*, 10 (3), 283-307
- La Casa, E., Lorente, E. (1999). Una experiencia sobre como responden los alumnos y alumnas ante una educación emancipadora. *I Congreso Internacional de educación física*. Fondo Internacional de Enseñanza
- Madrona, P. (2003). Interés y valoración de la educación por padres y alumnos en la enseñanza obligatoria. *Revista de Educación* 332 (23), 327-355
- M. de E. Nacional. (2006). *Orientaciones generales para la educación en tecnología* 6, 99
- Moreno, J. (2014). *El pensamiento computacional*. Proyecto Programamos. Accesible en: <http://programamos.es/faq/>
- Orengo, R. (2014). Estereotipos Sexuales y su relación con las conductas. *Rev. Puertorriqueña Psicología*, 23 (2), 48-61
- Ortiz, Elena. (2007). *Inteligencias múltiples en la educación de la persona*. (6ª Ed.) Argentina: Ediciones Bonum
- Pane J. (1998). "Designing a Programming System for Children with a Focus on Usability", in *Adjunct Proceedings SIGCHI'98: Conference Summary: Human Factors in Computing Systems*, Los Angeles, CA, pp. 62-63, 1998

Sarrió, M. (2002). *La Psicología de Género a través del “Techo de Cristal”*. Tesis Doctoral. Universitat de València. Dirigida por Ester Barberá. Manuscrito no publicado

Tezanos, J. F. (2001) “Hacia un nuevo paradigma social. La emergencia de las sociedades tecnológicas avanzadas”. *La sociedad dividida. Estructuras de clases y desigualdades en las sociedades tecnológicas*. Madrid, Biblioteca Nueva.

Waksman, N. (2005). *El papel de la mujer en la ciencia*. Universidad Autónoma de Nuevo León, México: Ciencia UANL

Wing, J. (2006). Computational thinking. *Communications of the ACM*, 49(3) , 33-35

Zúñiga, R., Hurtado, J. (2015). *Explorando los procesos de abstracción computacional en niños: identificando el uso de modelos mentales compartidos*. Proyecto de Tesis de Maestría en Universidad del Cauca