

Estudio de la influencia del aprendizaje del pensamiento computacional en las materias de ciencias en alumnos de secundaria



BOIX-TORMOS, JUAN-JOSE

TFM

Universitat Oberta de Catalunya

Tesis presentada para el grado de
Máster Universitario en Aplicaciones Multimedia

2016 Junio

1. Profesor: Enrique Guaus Termens

2. Profesor: David García Solórzano

Día de la defensa: 19/05/2016

Firma del presidente del tribunal:

Resumen

Los alumnos que actualmente cursan secundaria se encontrarán, al término de sus estudios, un mercado laboral, una sociedad y, en definitiva, un futuro muy diferente al actual. Debemos prepararlos para que sean capaces de superar estos nuevos retos. A los alumnos de secundaria, en concreto a los estudiantes de ciencias podemos dotarles de nuevas habilidades que les faciliten el estudio de las distintas materias de ciencias. Esto implica la necesidad de cambiar las estrategias de aprendizaje. Una de estas estrategias destaca en la actualidad, el Pensamiento Computacional. Este estudio aporta datos para verificar si el pensamiento computacional es útil, no sólo directamente en las ciencias de la computación; sino que, su conocimiento aporta beneficios al resto de materias de ciencias. La presente línea de investigación aporta experiencias y evalúa cómo influye en los resultados académicos de las asignaturas de ciencias, de aquellos alumnos de secundaria que han estado estudiando en la asignatura de informática pensamiento computacional a diferencia de aquellos que no lo han estudiado.

A mi familia

Reconocimientos

Me gustaría dar las gracias a todos los profesores que me han guiado hasta este momento.

Contenido

Lista de figuras	v
Lista de tablas	vii
Glosario	ix
1 Introducción	1
1.1 Justificación	1
1.2 Hipótesis	2
1.3 Preguntas de investigación	2
1.4 Metodología de investigación	2
1.4.1 Estrategia de investigación	2
1.4.2 Técnicas de generación de datos	4
1.5 Plan de investigación	5
2 Fundamento teórico	9
2.1 ¿Qué es el Pensamiento Computacional?	9
2.2 ¿Cómo se enseña el Pensamiento Computacional?	12
2.2.1 Pensamiento Computacional "sin programar"	12
2.2.2 Pensamiento Computacional "a través de la programación"	13
3 Estado del arte	15
3.1 Enseñar Pensamiento Computacional "sin programar"	15
3.1.1 CS Unplugged	15
3.2 Enseñar Pensamiento Computacional "a través de la programación"	16
3.2.1 SCRATH + ABP, como estrategia para el desarrollo del PC	16

CONTENIDO

3.2.2	Pensamiento Computacional a través de la Programación: Paradigma de Aprendizaje	17
3.2.3	¿Puede App Inventor aportar PC a K-12?	18
4	Experimento: Prototipo para la investigación. (Estudio previo)	21
4.1	Estudio previo	21
4.2	Hipótesis	21
4.3	Preguntas de investigación	21
4.4	Metodología de investigación	22
4.4.1	Estrategia de investigación	22
4.4.2	Tests	23
4.4.3	Población	24
4.4.4	Formación	26
4.5	Análisis estadístico	28
4.5.1	Estudio estadístico de la población	28
4.5.2	Resultados del experimento	33
4.5.3	Limitaciones del estudio y recomendaciones	36
5	Conclusiones y trabajo futuro	37
5.1	Conclusiones	37
5.2	Trabajo futuro	38
	Apéndices	39
.1	Apéndice I	41
.2	Apéndice II	44
	Referencias	47

Lista de figuras

1.1	Cronograma con el detalle de las tareas y la estimación en tiempo para la realización del estudio	7
1	Conjunto de datos tratados con SPSS	45

LISTA DE FIGURAS

Lista de tablas

1.1	Materias en Secundaria Obligatoria	3
1.2	Materias en las Modalidades de 1º de Bachillerato	3
1.3	Materias en las Modalidades de 2º de Bachillerato	4
2.1	Comparativa. Definiciones de Pensamiento Computacional	11
2.2	Habilidades que definen el Pensamiento Computacional	11
4.1	Asignaturas de 1º de Bachillerato consideradas en el estudio	24
4.2	Distribución de alumnos por grupos: Experimental, de Control y Descartados	25
4.3	Grupos de alumnos participantes en el estudio	26
4.4	Alumnos participantes por grupo y sexo	26
4.5	Variables del estudio	28
4.6	Variables del estudio en SPSS	28
4.7	Estadísticos descriptivos de la variable “NM2E: Nota media 2ª evaluación”	29
4.8	Prueba de Normalidad Shapiro Wilk (Var.NM2E)	30
4.9	Prueba Leneve (Var. “NM2E: Nota media 2ª evaluación”)	30
4.10	Prueba t de la variable “NM2E: Nota media 2ª evaluación”	31
4.11	Estadísticos descriptivos de la variable “PRET: Resultados PRE-TEST”	32
4.12	Prueba t de la variable “PRET: Resultados PRE-TEST”	32
4.13	Estadísticos descriptivos de la variable “NM3E: Nota media 3ª evaluación”	33
4.14	Prueba de Normalidad Shapiro Wilk (Var.NM3E)	34
4.15	Prueba Leneve (Var. “NM3E: Nota media 3ª evaluación”)	34
4.16	Prueba t de la variable “NM3E: Nota media 3ª evaluación”	35
4.17	Comparación: “Notas medias 2ª y 3ª evaluación”	35

GLOSARIO

Glosario

CT Computational Thinking

PC Pensamiento computacional

CSTA Computer Science Teachers Association

ISTE International Society for Technology Education

NCS National Science Foundation

GLOSARIO

1

Introducción

1.1 Justificación

Los alumnos que actualmente cursan secundaria se encontrarán, al término de sus estudios, un mercado laboral, una sociedad y, en definitiva, un futuro muy diferente al actual. Debemos prepararlos para que sean capaces de superar estos nuevos retos. Uno de ellos será unir el conocimiento humano con las nuevas tecnologías que tendrán que utilizar. A los alumnos de secundaria, en concreto a los estudiantes de ciencias podemos dotarles de nuevas habilidades que les faciliten el estudio de las distintas materias de ciencias. Para enseñar a los estudiantes estas nuevas competencias existe la necesidad de cambiar las estrategias de aprendizaje.

Entre las nuevas estrategias que están destacando surge el pensamiento computacional. En la actualidad, cada vez más, nos encontramos con la omnipresencia del pensamiento computacional (PC) en nuestra vida (1)(2).

Para Wing (3)(4), el pensamiento computacional “implica la resolución de problemas, diseño de sistemas, y la comprensión de la conducta humana, haciendo uso de los conceptos fundamentales de la informática. El pensamiento computacional incluye una serie de herramientas mentales que reflejan la amplitud del campo de la informática”.

Según Grover y Pea (5) hay mucha literatura relacionada con el pensamiento computacional, pero hay poca relacionada con experiencias con alumnos de primaria y de secundaria. También está poco descrita la posible mejora que, el aprendizaje del

1. INTRODUCCIÓN

pensamiento computacional, aporta en el resto de materias, especialmente en las de ciencias, o en la misma vida.

Este estudio pretende aportar datos para verificar si el pensamiento computacional es útil, no sólo directamente en las ciencias de la computación; sino que, su conocimiento aporta beneficios al resto de materias de ciencias.

Más concretamente, la línea de investigación aporta experiencias y evalúa cómo influye en los resultados académicos de las asignaturas de ciencias, de aquellos alumnos de secundaria que han estado estudiando en la asignatura de informática pensamiento computacional a diferencia de aquellos que no lo han estudiado.

1.2 Hipótesis

El aprendizaje del Pensamiento Computacional influye en los resultados académicos en las materias de ciencias en alumnos de secundaria.

1.3 Preguntas de investigación

- ¿En qué medida la mejora de las habilidades descritas por el PC implica una mejora en los resultados académicos en las materias de ciencias en alumnos de secundaria?
- ¿De las diferentes habilidades en que se puede descomponer el PC hay alguna que determine mejor la influencia que propone la hipótesis?
- ¿Influyen las características personales del sujeto (Ej.: género)?

1.4 Metodología de investigación

1.4.1 Estrategia de investigación

Con el objetivo de aportar datos medibles que denoten correlación entre el aprendizaje del PC y la posible influencia de este aprendizaje en materias de ciencias, la estrategia de investigación es un estudio que proporciona datos que permiten un tratamiento informático para su tratamiento estadístico.

El estudio se desarrolla en Institutos de Educación Secundaria de la Comunidad Valenciana, donde se imparte la asignatura informática. En la Comunidad Valenciana

1.4 Metodología de investigación

la asignatura informática es una asignatura optativa de oferta obligatoria desde 2007 (6). Los niveles donde se desarrolla el estudio abarcan toda la secundaria obligatoria desde 1º de ESO hasta 4º de ESO y en 1º y 2º de bachillerato.

Dada la gran variedad de itinerarios durante los cursos de secundaria que se establecen en el DECRETO 87/2015, de 5 de junio (7), se ha optado por las siguientes materias de ciencias sobre las que se realiza la observación académica en cada uno de los cursos. En la tabla 1.1 se detallan las asignaturas en los cursos 1º a 4º de ESO.

1º ESO	2º ESO	3º ESO	4º ESO
Biología y Geología	Biología y Geología	Biología y Geología	Biología y Geología
Física y Química	Física y Química	Física y Química	Física y Química
Matemáticas	Matemáticas	Matemáticas Orientadas a las Enseñanzas Académicas	Matemáticas Orientadas a las Enseñanzas Académicas
		Matemáticas Orientadas a las Enseñanzas Aplicadas	Matemáticas Orientadas a las Enseñanzas Aplicadas

Table 1.1: Materias en Secundaria Obligatoria

En 1º y 2º de bachillerato, solamente se estudian las modalidades de Ciencias y Ciencias sociales. En las tablas 1.2 1.3 se relacionan las asignaturas que se estudian.

Ciencias	Ciencias Sociales	Humanidades	Artes
Matemáticas I	Matemáticas Aplicadas a las Ciencias Sociales I	Ninguna	Ninguna
Física y Química			
Biología y Geología			

Table 1.2: Materias en las Modalidades de 1º de Bachillerato

1. INTRODUCCIÓN

Ciencias	Ciencias Sociales	Humanidades	Artes
Matemáticas II	Matemáticas Aplicadas a las Ciencias Sociales II	Ninguna	Ninguna
Física			
Química			
Biología			
Geología			

Table 1.3: Materias en las Modalidades de 2° de Bachillerato

1.4.2 Técnicas de generación de datos

Para el desarrollo del estudio se utilizan las siguientes técnicas de generación de datos:

1. Entrevista con los profesores responsables. Nos ayudan a conocer el tipo de formación que han adquirido los alumnos. Y nos aportan información de como ha desarrollado el proceso de formación.
2. Test (pre-post formación en PC). Para medir la capacidad de pensamiento computacional de un sujeto (8)(9). Necesitamos cuantificar esta medida antes y después de iniciar a los alumnos con el pensamiento computacional.
3. Recopilación de las calificaciones académicas de la segunda y tercera evaluación de las materias de ciencias. Conociendo los resultados académicos de los sujetos de estudio en estos dos momentos (pre-post), es posible cuantificar si se ha producido alguna mejora.

El tipo de análisis de datos es cuantitativo para los datos medibles que nos proporcionarán los tests y la recopilación de calificaciones académicas. En cambio, es cualitativo para la entrevista con los profesores.

1.5 Plan de investigación

Tareas realizadas durante la investigación:

T.0. PROPUESTA DE INVESTIGACIÓN

Durante la propuesta de investigación se ha obtenido una visión general del estudio, se ha realizado una primera revisión de la literatura para centrar mejor el ámbito de conocimiento y los objetivos iniciales de la investigación.

T.1. ESTADO DEL ARTE

Revisión exhaustiva de los estudios previos sobre el objeto de investigación determinando los siguientes objetivos:

T.1.1. ¿Qué es el Pensamiento Computacional?

T.1.2. ¿Cómo se enseña el Pensamiento Computacional?

T.1.3. ¿Qué resultados ofrecen los estudios previos?

T.2. DISEÑO DE FORMULARIOS

Se han diseñado primero las variables objeto de estudio. Por una parte se obtienen datos académicos de los alumnos y por otro los resultados de los tests pre y post formación en PC.

Se ha realizado un tratamiento específico, de manera que se preserve la confidencialidad de los sujetos de estudio. Existe una relación unívoca entre el resultado académico del alumno X y los resultados de sus test. Pero por otra parte no es necesario identificar personalmente al alumno. Se ha realizado un sistema de asociación que se ha comunicado al profesor colaborador. Este es el único que puede realizar la asociación personal.

En segundo lugar se ha diseñado la entrevista con los profesores que realizarán la formación. La entrevista y los tests se van mejorando con la puesta en práctica.

T.3. DISEÑO DEL ÁMBITO DE ESTUDIO

La propuesta de estudio intenta ser lo más amplia posible abarcando el mayor número de Institutos de Educación Secundaria. A partir de los listados oficiales de IES de la región se ha contactado con ellos para comprobar que se estudia la materia de informática.

Se ha acordado con los profesores la realización de las pruebas y los periodos en los que se realizará la formación. De este estudio se ha obtenido la lista definitiva de IES

1. INTRODUCCIÓN

en los que se puede desarrollar el estudio. Se ha acordado realizar un estudio piloto en un solo centro.

T.4. DESARROLLO DEL ESTUDIO

Con la entrevista al profesor colaborador se obtienen datos demográficos como edad, género y datos relativos a la investigación como sus conocimientos previos en PC, experiencia previa en la formación, etc.

Se obtienen los datos académicos previos de los alumnos; es decir, los relativos a la 2ª evaluación. Se realizaron los test iniciales a todos los alumnos de los grupos participantes en el estudio, incluso a aquellos alumnos que no recibieron formación en PC. Estos alumnos componen el grupo de control.

Durante el periodo de formación, 3ª evaluación, se han realizado visitas de seguimiento con el objeto de obtener información adicional que aporte nuevas perspectivas al estudio.

Durante el mes de junio se han realizado los tests finales posteriores a la formación. Y para completar la recolección de datos se han obtenido los datos relativos a la 3ª evaluación de los alumnos.

T.5. ESTUDIO ESTADÍSTICO

El estudio estadístico consiste en un tratamiento informático de los datos con el programa SPSS. Finalmente se realiza el contraste de hipótesis mediante pruebas estadísticas.

T.6. CONCLUSIONES

Redacción de la memoria de tesis con las conclusiones del estudio.

T.7. DEFENSA / LECTURA DE LA TESIS

Defensa y lectura de la Tesis.

1.5 Plan de investigación

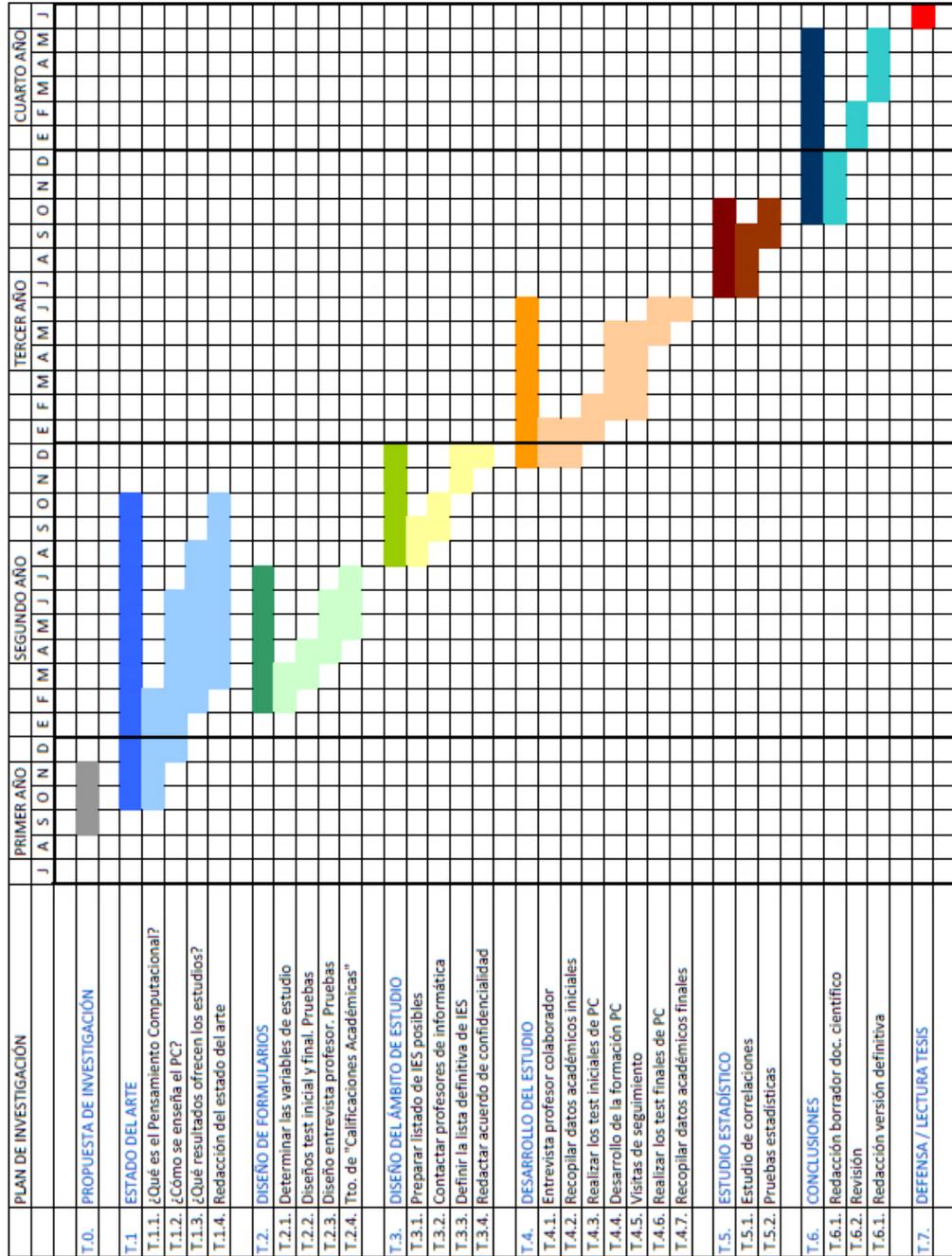


Figure 1.1: Cronograma con el detalle de las tareas y la estimación en tiempo para la realización del estudio

1. INTRODUCCIÓN

2

Fundamento teórico

Este capítulo aporta el estado actual del discurso sobre el pensamiento computacional entorno a las experiencias de la enseñanza de la programación en secundaria, con el objetivo de crear un marco de referencia para esta y futuras investigaciones en esta área.

En primer lugar se describe cómo se define actualmente el pensamiento computacional. En el siguiente apartado se identifican los principales métodos que se están utilizando en secundaria para la enseñanza del PC.

2.1 ¿Qué es el Pensamiento Computacional?

Según Jannet Wing (2006) (3), "pensamiento computacional implica la resolución de problemas, diseño de sistemas, y la comprensión de la conducta humana, haciendo uso de los conceptos fundamentales de la informática. Pensamiento computacional incluye una serie de herramientas mentales que reflejan la amplitud del campo de la informática".

Según Hemmendinger (2010) (10), la mayoría de las definiciones de pensamiento computacional que se ofrecen actualmente carecen de precisión y no proporcionan suficientes ejemplos, a menudo haciendo el concepto mal entendido o mal interpretado.

Jannet Wing (2010) (4) revisó la definición, "pensamiento computacional es el proceso de pensamiento involucrado en la formulación de los problemas y sus soluciones para que estas soluciones están representadas de forma que se puedan llevar a cabo con eficacia por un agente de procesamiento de información." Aho (2012) (11)

2. FUNDAMENTO TEÓRICO

simplifica la definición de PC como “los procesos mentales implicados en la formulación de los problemas por lo que sus soluciones pueden ser representadas como pasos computacionales y algoritmos”.

Weinberg (2013) (12) cita dos definiciones como destacadas actualmente. La primera una definición operativa publicada por el CSTA en colaboración con la Sociedad Internacional para la Tecnología en la Educación (ISTE) líderes en la educación y la industria (2011) (13). La segunda ofrecida por la iniciativa “Explorando el Pensamiento Computacional” de Google, el primer programa a gran escala para proporcionar una definición operativa, la difusión de los recursos y promover la discusión educadores sobre el pensamiento computacional (2010) (14).

La primera definición describe el pensamiento computacional como un proceso de resolución de problemas que incluye, pero no se limita a, las siguientes características:

(A.1) Formulación de problemas de una manera que nos permite utilizar una computadora y otras herramientas para ayudar a resolverlos.

(A.2) Organizar y analizar los datos lógicamente.

(A.3) Representar los datos a través de abstracciones tales como modelos y simulaciones.

(A.4) La automatización de soluciones a través del pensamiento algorítmico (una serie de pasos ordenados).

(A.5) Identificar, analizar e implementar posibles soluciones con el objetivo de lograr la combinación más eficiente y eficaz de las medidas y los recursos.

(A.6) Generalizando la transferencia de este proceso de resolver problemas a una amplia variedad de problemas.

En la segunda definición, Google describe un proceso que incluye cuatro técnicas de pensamiento computacional: La descomposición, reconocimiento de patrones, generalización y abstracción, y diseño de algoritmos (14).

(B.1) Descomposición: Dividir un problema en partes o pasos.

(B.2) Reconocer y encontrar patrones o tendencias.

(B.3) Generalizar patrones y tendencias en las normas, principios o ideas.

(B.4) Diseño de algoritmos: Desarrollar las instrucciones para resolver un problema o pasos de una tarea.

En la tabla 2.1 se muestra una comparativa de las dos definiciones. Ambas se refieren, con matices, a las mismas habilidades.

2.1 ¿Qué es el Pensamiento Computacional?

CSTA / ISTE	Google
(A.1) Usar ordenadores para resolver problemas	(B.1) Descomposición
(A.2) Organizar datos lógicamente	(B.2) Patrones y tendencias
(A.3) Abstracciones, modelos y simulaciones	(B.3) Generalizar patrones
(A.4) Pensamiento algorítmico	
(A.5) Buscar soluciones eficientes y eficaces	(B.4) Desarrollar algoritmos
(A.6) Generalizar las soluciones	

Table 2.1: Comparativa. Definiciones de Pensamiento Computacional.

A partir de las de definiciones anteriores y con el objetivo de poder evaluar el conjunto de habilidades que definen el pensamiento computacional definimos el pensamiento computacional como el siguiente conjunto de habilidades: análisis, abstracción y secuenciación. En la tabla 2.2 se detallan estas habilidades.

Habilidades	
(C.1) Análisis	Habilidad para encontrar patrones y tendencias.
(C.2) Abstracción	Uso de la abstracción para la representación de los datos.
(C.3) Secuenciación	Habilidad para dividir y estructurar ordenadamente los pasos en los que se divide un problema.

Table 2.2: Habilidades que definen el Pensamiento Computacional.

Habilidad: (C.1) Análisis

Definimos análisis en PC como la habilidad para encontrar patrones y tendencias. El desarrollo de la capacidad de análisis ante un problema dado es fundamental puesto que conduce a fijar el trabajo de resolución. Ha de facilitar la interpretación de la información que se posee para su resolución y permitir inferir información deduciendo patrones y tendencias en los datos aportados.

Habilidad: (C.2) Abstracción

Jannet Wing (2010) (4), considera que “la abstracción es el proceso de pensamiento más importante y de alto nivel en el pensamiento computacional”. La abstracción, dice Wing, “se utiliza para capturar las propiedades esenciales comunes a un conjunto de objetos al tiempo que oculta las distinciones irrelevantes entre ellos”. Concluye que “la abstracción nos da el poder para escalar y hacer frente a la complejidad”.

2. FUNDAMENTO TEÓRICO

Habilidad: (C.3) Secuenciación

Esta habilidad comprende la definición de algoritmos. Un algoritmo define una serie de pasos ordenados que se han de seguir para resolver un problema. Los algoritmos se pueden representar de varias formas, en lenguaje natural, pseudocódigo, diagramas de flujo o lenguajes de programación.

2.2 ¿Cómo se enseña el Pensamiento Computacional?

Existen principalmente dos puntos de vista a la hora de afrontar la enseñanza del PC. El primero no hace uso inicial de la programación y el segundo comienza precisamente programando y dejando que el aprendizaje se realice a través de la experiencia.

2.2.1 Pensamiento Computacional "sin programar"

Guzdial, M (2008) (15) afirma que "Enseñar el pensamiento computacional a todo el mundo puede requerir enfoques diferentes que los que usamos cuando asumimos que nuestros estudiantes quieren convertirse en profesionales de la informática".

Así, en el primer punto de vista según James J. Lu (2009) (16) la programación no ha de ser el primer paso pues hay una serie de conceptos y aprendizajes previos que han de conocer los estudiantes. Para James J. Lu debemos separar la enseñanza del pensamiento computacional de la programación en sí.

James J. Lu (16) sostiene que "la enseñanza debe centrarse en establecer vocabularios y símbolos que se pueden utilizar para anotar y describir la computación y la abstracción, sugerir información y ejecución, y proporcionar la notación en torno al cual los modelos mentales de los procesos se pueden construir".

También afirma que la complejidad de los lenguajes de programación son de nivel superior a las habilidades que aporta el pensamiento computacional. Así, para enseñar estas habilidades útiles en otras disciplinas distintas a la informática "tenemos que poner el énfasis en la comprensión (y ser capaz de realizar de forma manual) procesos computacionales, y no en sus manifestaciones en los lenguajes de programación específicos".

Esta forma de enseñar pensamiento computacional pretende que los estudiantes se familiaricen con los conceptos algorítmicos como flujo básico de control y desarrollen habilidades para la abstracción y la presentación de información, y para la evaluación

2.2 ¿Cómo se enseña el Pensamiento Computacional?

de las propiedades de los procesos, mediante el aprendizaje de un vocabulario y lenguaje de pensamiento computacional.

2.2.2 Pensamiento Computacional ”a través de la programación”

En el segundo punto de vista, la enseñanza del PC, se basa en el uso de entornos y herramientas que fomenten el PC. Son muy populares los entornos de programación Scratch (17) y App Inventor (18) desarrollados por el MIT. Este último permite de manera relativamente sencilla crear aplicaciones para dispositivos móviles. Ambos están basados en permitir una fácil incorporación de usuarios que no estén familiarizados en programación pero que al mismo tiempo permiten desarrollos de elevada complejidad.

Estos entornos cumplen con los requisitos que cita Seymour Papert (19) Los lenguajes de programación han de tener un “suelo bajo” (fácil empezar a programar) y un “techo alto” (oportunidades de crear proyectos más complejos con el tiempo).

2. FUNDAMENTO TEÓRICO

3

Estado del arte

En este apartado se presentan algunas experiencias de enseñanza del pensamiento computacional en alumnos de secundaria. La primera experiencia sigue el punto de vista enseñar Pensamiento Computacional "sin programar". En las siguientes se sigue el punto de vista donde se enseña el pensamiento computacional "a través de la programación".

3.1 Enseñar Pensamiento Computacional "sin programar"

3.1.1 CS Unplugged

En este apartado se describe la experiencia desarrollada por el Grupo de Investigación "Educación en Ciencias de la Computación" de la Universidad de Canterbury, Nueva Zelanda denominado "CS Unplugged". (20)

"CS Unplugged" es un conjunto de actividades de aprendizaje gratuitos desarrollados para enseñar Ciencias de la Computación. Está patrocinado por la empresa Google y se ofrece bajo una licencia Creative Commons BY-NC-SA, por lo que se puede copiar, compartir y modificar.

El objetivo principal del proyecto es promover las Ciencias de la Computación (y la informática en general) a los jóvenes como una disciplina interesante, atractiva y estimulante intelectualmente. Se pretende transmitir una serie de fundamentos que no dependen de un software o sistema en particular, conceptos que permanecerán frescos después de 10 años.

3. ESTADO DEL ARTE

Aunque se presentan los conceptos fundamentales en Ciencias de la Computación, como algoritmos, inteligencia artificial, gráficos, teoría de la información, interfaces hombre-ordenador y lenguajes de programación, las actividades propuestas no dependen de los ordenadores, con el objetivo de evitar confundir Ciencias de la Computación con la programación o el aprendizaje de software de aplicación. Tratan pues de exponer las ideas sin tener que superar primero la barrera de aprender a programar. Afirman que: la programación es un medio, no un fin. Los conceptos se enseñan mediante actividades del tipo "aprender-haciendo", también se potencia el trabajo en equipo y la comunicación de los resultados. El enfoque es constructivista y permite que los estudiantes lleguen a las soluciones por si mismos.

Las actividades son divertidas y atractivas. Por lo general, las explicaciones son bastante breves (el profesor expone los materiales y unas pocas reglas), son los estudiantes los que alcanzan los objetivos. Para el desarrollo de las actividades no se necesita ningún equipo especializado, los materiales necesarios son de bajo coste. Las actividades se planean como módulos independientes y están preparadas de manera que si se cometen algunos errores se pueda alcanzar igualmente los objetivos.

Este proyecto está apoyado y promovido por diversas organizaciones y empresas de todo el mundo: ACM K-12 y los comités curriculares; CSTA, programa de Asociación de Maestros de Ciencias de la Computación; CS4ALL (UW, CMU y UCLA); NCWIT (Centro Nacional de EE.UU. para Mujeres y Tecnología de la Información); TECS (Teacher Enrichment in Computer Science); Google y Microsoft.

3.2 Enseñar Pensamiento Computacional "a través de la programación"

3.2.1 SCRATH + ABP, como estrategia para el desarrollo del PC

El proyecto plantea una estrategia metodológica basada en el uso del programa Scratch + ABP (Aprendizaje Basado en Problemas), para el desarrollo de competencias de pensamiento computacional. (21)

Scratch (17) se basa en las figuras de bloques de construcción. Estos bloques son autoencajables cuando son sintácticamente correctos. Los procedimientos se construyen encajando los bloques, fácilmente, mediante arrastrar y soltar.

3.2 Enseñar Pensamiento Computacional ”a través de la programación”

El Aprendizaje Basado en Problemas (ABP), está enfocado en estimular al estudiante a solucionar problemas reales que se le planteen y que en grupos de trabajo pequeños, colaborativamente, busquen dar solución a éstos. Es una estrategia donde el estudiante es el foco central y el docente el facilitador del proceso de aprendizaje.

La investigación utilizó un enfoque descriptivo, cualitativo y cuantitativo, comparando los resultados de las notas finales de los años 2013 y 2014 de 162 alumnos de edades entre los 11 y 13 años, en el área de tecnología e informática, comparando estos resultados con el desarrollo de las competencias de pensamiento computacional.

Los instrumentos utilizados fueron la observación, encuestas, las rúbricas de evaluación, notas y la documentación del diario de campo. Se trató de incentivar en el estudiante un aprendizaje significativo y autónomo, que le permita pensar, antes de actuar a partir de un escenario propuesto.

Se planteó un marco de trabajo para integrar en un ambiente de aprendizaje la estrategia de ABP y el uso de SCRATCH para elevar los bajos resultados de los estudiantes alcanzados en el año 2013.

Posteriormente, en el año 2014, al evaluar nuevamente las competencias se obtuvo que el 60% de los estudiantes reflejaron una mejoría del rendimiento académico. De las encuestas finales a los estudiantes se comprobó que el uso del programa educativo SCRATCH, como mediador para dar solución a problemas planteados, permite que el estudiante potencie ciertas habilidades en un escenario lúdico. Los estudiantes consideran que les obliga a pensar antes de crear y recrear sus ideas.

3.2.2 Pensamiento Computacional a través de la Programación: Paradigma de Aprendizaje

Este artículo (22) presenta el concepto del Pensamiento Computacional y cómo puede ser integrado en el aula a través del diseño e implementación de proyectos de programación. Además en la última sección del artículo se muestra el contenido y los resultados del curso “Pensamiento Computacional en la Escuela” impartido en la modalidad MOOC (Massive Open Online Courses) en la plataforma Miríada X.

La experiencia se centra en los conceptos del PC: abstracción, descomposición y generalización. Los dos primeros conceptos, abstracción y descomposición se desarrollan a partir del conocido juego PONG como herramientas para resolver tareas complejas. El juego consiste en: ”Una bola azul rebota en las paredes y en la paleta rectangular

3. ESTADO DEL ARTE

negra que está controlada por el jugador. Un marcador lleva la cuenta del número de veces que el jugador ha golpeado la bola. Finalmente, si la paleta no logra golpear la bola y ésta llega a la línea roja del fondo el juego termina.”

La abstracción se concreta en identificar la esencia del proceso eliminando lo accesorio. Se reducen los elementos del juego para conservar sólo los elementos esenciales. En cuanto a la descomposición, se divide el problema en partes: movimiento de la bola, movimiento de la paleta, marcador.

El tercer concepto, la generalización, se aprende a través de una simplificación del juego Pack-Man.

”En la implementación del juego no hay enemigos del Pack-Man y el objetivo es llegar a través del laberinto al cuadrado rojo lo más rápido posible. Se puede utilizar un reloj para medir la habilidad de cada jugador registrando el tiempo que requiere para alcanzar el objetivo. Una vez que se alcanza el objetivo, el diseño del laberinto cambia, y empieza un nuevo escenario del juego”.

Se proponen ejemplos de generalizaciones, como la que se plantea en la creación de juegos o la animación de relatos sencillos en la que nos encontramos con la necesidad de ”programar scripts” que implementen la música de fondo. Otro ejemplo propuesto es la animación del Pack-Man o el guiado con los cursores.

3.2.3 ¿Puede App Inventor aportar PC a K-12?

El proyecto (23) explora las formas de utilizar App Inventor para introducir el pensamiento computacional a maestros y estudiantes de secundaria. App Inventor es una nueva plataforma de programación visual para crear aplicaciones móviles para teléfonos inteligentes basados en Android. Tiene un gran potencial para atraer a una nueva generación de estudiantes a la informática y el pensamiento computacional.

Los estudiantes se enfrentaron a App Inventor por su cuenta y solo tuvieron el apoyo de los tutoriales en línea de Google. Aún así, los estudiantes, aseguraron que aprender a programar es una actividad agradable y creativa que ayuda a mejorar la capacidad para resolver problemas y satisfacer desafíos.

En las conclusiones finales el autor asegura que App Inventor ha demostrado ser por una parte fácil de usar y al mismo tiempo potente.

El estudio concluye que App Inventor está indicado para que los estudiantes se centren en la resolución de los problemas más que en aprender la sintaxis de un lenguaje.

3.2 Enseñar Pensamiento Computacional "a través de la programación"

Destaca la motivación del estudiante incluso más que otras aplicaciones similares como SCRATCH o ALICE, "por su naturaleza portátil y su utilidad", pues se basa en el uso de los móviles, algo importante en la vida de los estudiantes.

Para la formación App Inventor tiene el soporte de Google. En particular, el grupo de App Inventor Google supone un enorme apoyo.

Finalmente el estudio concluye que entre los profesores asistentes hubo un consenso general que App Inventor tiene un enorme potencial para el futuro de las escuelas secundarias.

3. ESTADO DEL ARTE

4

Experimento: Prototipo para la investigación. (Estudio previo)

4.1 Estudio previo

El estudio se desarrolla con un número limitado de alumnos. Este prototipo desarrolla toda la estrategia necesaria para llevar a cabo el estudio con una población mayor. Con este prototipo se pretende poner en práctica la estrategia de investigación, detectar y eliminar errores; así como posibles mejoras.

Las hipótesis y preguntas de investigación son las mismas que en el estudio inicial planteado. Los resultados obtenidos tendrán un valor limitado por hacer uso de una población pequeña.

4.2 Hipótesis

El aprendizaje del Pensamiento Computacional influye en los resultados académicos en las materias de ciencias en alumnos de secundaria.

4.3 Preguntas de investigación

- ¿En qué medida la mejora de las habilidades descritas por el PC implica una mejora en los resultados académicos en las materias de ciencias en alumnos de secundaria?

4. EXPERIMENTO: PROTOTIPO PARA LA INVESTIGACIÓN. (ESTUDIO PREVIO)

- ¿De las diferentes habilidades en que se puede descomponer el PC hay alguna que determine mejor la influencia que propone la hipótesis?
- ¿Influyen las características personales del sujeto (Ej.: género)?

4.4 Metodología de investigación

4.4.1 Estrategia de investigación

Con el objetivo de aportar datos medibles que denoten correlación entre el aprendizaje del PC y la posible influencia de este aprendizaje en materias de ciencias, la estrategia de investigación desarrollada es un experimento que proporciona datos que mediante un tratamiento informático permiten un tratamiento estadístico.

El estudio se ha desarrollado en un Instituto de Educación Secundaria de la Comunidad Valenciana donde se oferta la asignatura optativa informática. El estudio se ha reducido a los estudiantes de primer curso de bachillerato.

En el estudio nos encontramos con algunos condicionantes:

La formación de los grupos no ha sido aleatoria. Nos encontramos con una situación previa al inicio de la investigación. La pertenencia a un grupo viene determinada por el análisis de las características (asignaturas en las que se ha matriculado) de cada individuo.

El experimento no es reproducible en un laboratorio. Existen múltiples variables que afectan al estudio y que no podemos controlar ni medir, por ejemplo: La formación se desarrolla durante la tercera evaluación, momento en el que los estudiantes pueden experimentar un incremento de motivación al ver cerca la finalización del curso. Este punto dificulta nuestra capacidad de descartar explicaciones alternativas.

”Un estudio es cuasi-experimental cuando no hay un control efectivo de las variables de selección” (24). Por tanto la estrategia esta basada en una metodología cuasi-experimental pre-post con grupo de cuasi-control. (25)

El tipo de análisis de datos es cuantitativo: los datos de las puntuaciones de los tests y las calificaciones académicas obtenidas en las evaluaciones 2^a y 3^a.

4.4.2 Tests

Para medir la evolución del pensamiento computacional de un sujeto (8)(9), necesitamos cuantificar esta medida antes de iniciar a los alumnos con el pensamiento computacional y al término de esta formación.

El nivel de los tests, pre y post, es de nivel similar. Para ello y para evitar obtener puntuaciones sesgadas por la dificultad intrínseca de las preguntas se han diseñado suficientes cuestiones que se asignan aleatoriamente a los estudiantes. Los tests están adaptados al nivel de los alumnos.

Para la definición de los tests se ha tenido en cuenta la evaluación de cada una de las habilidades propuestas. En el Apéndice .1 se proporciona el que se ha utilizado con los estudiantes de 1º de bachiller.

1. Habilidad: (C.1) Análisis

Definimos análisis en PC como la habilidad para encontrar patrones y tendencias. El desarrollo de la capacidad de análisis ante un problema dado es fundamental puesto que conduce a fijar el trabajo de resolución. Ha de facilitar la interpretación de la información que se posee para su resolución y permitir inferir información deduciendo patrones y tendencias en los datos aportados. Corresponde con las cuestiones 1 y 2 del Apéndice .1.

2. Habilidad: (C.2) Abstracción

Jannet Wing (2011) (4), considera que “la abstracción es el proceso de pensamiento más importante y de alto nivel en el pensamiento computacional”. La abstracción, dice Wing, “se utiliza para capturar las propiedades esenciales comunes a un conjunto de objetos al tiempo que oculta las distinciones irrelevantes entre ellos”. Concluye que “la abstracción nos da el poder para escalar y hacer frente a la complejidad”.

Valoraremos la habilidad “abstracción” evaluando si el sujeto estudiado es capaz de reconocer los diferentes tipos de datos de la vida real que se plantean. También, se presenta la posibilidad de, ante un problema dado, capturar los datos que son relevantes e ignorar los que no lo son. Corresponde con las cuestiones 3 y 4 del Apéndice .1.

4. EXPERIMENTO: PROTOTIPO PARA LA INVESTIGACIÓN. (ESTUDIO PREVIO)

3. Habilidad: (C.3) Secuenciación

Esta habilidad comprende la definición de algoritmos. Un algoritmo define una serie de pasos ordenados que se han de seguir para resolver un problema. Los algoritmos se pueden representar de varias formas, en lenguaje natural, pseudocódigo, diagramas de flujo o lenguajes de programación.

Valoraremos la habilidad “secuenciación” evaluando si el sujeto es capaz de reconocer las secuencias adecuadas en cada caso. Corresponde con las cuestiones 5, 6 y 7 del Apéndice .1.

4.4.3 Población

En el instituto donde se ha realizado el estudio se pueden cursar dos tipos de bachillerato: Bachillerato de Ciencias (BAC) y Bachillerato de Ciencias Sociales y Humanidades (BACH). En la tabla 4.1 se detallan las asignaturas de ciencias que se cursan en cada modalidad. Los 36 alumnos de 1º de Bachiller cursan alguna de las dos modalidades que oferta el IES:

- BAC: Bachillerato de Ciencias (18 alumnos)
- BACH: Bachillerato de Ciencias Sociales y Humanidades (18 alumnos)

Los alumnos tienen edades entre 16 y 19 años, 24 de ellos son del sexo masculino y 12 del sexo femenino.

BAC: Ciencias	BACH: Ciencias Sociales y Humanidades
Matemáticas I	Matemáticas Aplicadas a las Ciencias Sociales I
Física y Química	
Biología y Geología	

Table 4.1: Asignaturas de 1º de Bachillerato consideradas en el estudio

Los alumnos pueden optar entre varias optativas una de ellas es la asignatura de informática denominada TIC I: Tecnologías de la Información y la Comunicación I.

Del total de 36 alumnos de primero de bachillerato consideramos los siguientes grupos:

- *Descartados*: Alumnos que cursan la modalidad BACH sin ninguna asignatura de ciencias ni TIC I. En la población estudiada existen 9 alumnos.
- *Experimental*: Alumnos de bachillerato que cursan asignaturas de ciencias y también han optado por TIC I. En este grupo nos encontramos con 12 alumnos de la modalidad BAC y 2 alumnos en la modalidad BACH.
- *Control*: Alumnos de bachillerato que cursan asignaturas de ciencias y NO han optado por TIC I. En este grupo nos encontramos con 7 alumnos de la modalidad BAC y 6 alumnos en la modalidad BACH.

En la tabla 4.2 se describe la distribución de alumnos en grupos. La nota media corresponde a la media de las notas obtenidas en la 2ª evaluación en las asignaturas de ciencias que ha cursado el alumno.

Modalidad	Asignaturas de Ciencias	Optativa TIC I	Nº Alumnos	Nota Media	Grupo
BAC	SI	SI	12	5,2	Experimental 1
BAC	SI	NO	6	3,8	Control 1
BACH	SI	SI	2	4,0	Experimental 2
BACH	SI	NO	7	5,4	Control 2
BACH	NO	NO	9	–	Descartados
			36		

Table 4.2: Distribución de alumnos por grupos: Experimental, de Control y Descartados

Se ha realizado un sistema de asociación acordado con el profesor colaborador. A cada alumno se le asigna un código con el fin de mantener el anonimato. El profesor es el único que puede realizar la asociación personal. Con el objetivo de preservar la confidencialidad de los sujetos de estudio, la relación entre el código de alumno y el nombre del alumno no será pública en ningún momento.

4. EXPERIMENTO: PROTOTIPO PARA LA INVESTIGACIÓN. (ESTUDIO PREVIO)

Teniendo en cuenta el número reducido de alumnos en este estudio, en la tabla 4.3 se indican los grupos que se consideran. En la tabla 4.4 se muestra la distribución de los alumnos por sexo con indicación de las notas medias.

Grupo	Asignaturas de Ciencias	Optativa TIC I	BAC	BACH	Nº Alumnos	Nota Media
Experimental	SI	SI	12	2	14	5,0
Control	SI	NO	6	7	13	4,7
Total			18	9	27	4,9

Table 4.3: Grupos de alumnos participantes en el estudio

Grupo	Nº Alumnos	Sexo masculino	Nota Media	Sexo femenino	Nota Media
Experimental	14	11	5,1	3	4,7
Control	13	9	4,6	4	4,8
Total	27	20	4,9	7	4,7

Table 4.4: Alumnos participantes por grupo y sexo

4.4.4 Formación

La formación se ha desarrollado durante los meses de mayo y junio. El profesor ha ido introduciendo los conceptos de lenguajes de programación de alto nivel a medida que se han ido realizando ejercicios prácticos de programación de nivel creciente. La formación ha seguido el punto de vista de enseñar PC a través de la programación.

El entorno de desarrollo utilizado ha sido GAMBAS con licencia GPL GNU/Linux. GAMBAS es una herramienta de desarrollo visual de aplicaciones parecida a los programas comerciales Microsoft Visual Basic o Delphi. GAMBAS (26) desarrollado inicialmente por Benoit Minisini, es el acrónimo de “Gambas Almost Means BASIC”, es decir, GAMBAS es casi como BASIC. Es un entorno de desarrollo integrado para la creación de todo tipo de aplicaciones sobre sistemas GNU/Linux y Windows. Basado en el lenguaje de programación BASIC, ofrece facilidades para generar aplicaciones de distinto nivel de complejidad de propósito general. Uno de los objetivos de GAMBAS es acercar el desarrollo de aplicaciones a personas no expertas en programación.

La selección de este entorno se ha basado en:

- La facilidad de instalación del entorno en los equipos del centro. El aula donde se desarrolla la formación está dotada de ordenadores de prestaciones medias con el sistema operativo UBUNTU 14.
- GAMBAS permite la creación de programas sencillos sin conocimientos iniciales de programación.
- Permite desarrollar programas bien estructurados y está preparado para desarrollos orientados a objetos.
- Existe una activa comunidad de desarrollo (27).

El tiempo dedicado a la programación ha sido de 18 horas (6 semanas a razón de 3 horas semanales). El alumnado, que hasta este momento no había tenido ningún contacto con la programación, ha estado muy interesado y se ha encontrado satisfecho con las pruebas de los programas realizados preguntando como podrían llegar a desarrollar programas de mayor complejidad.

El profesor afirma que los alumnos han alcanzado los siguientes objetivos:

- Conocer la necesidad de los lenguajes de alto nivel para el desarrollo de programas de ordenador.
- Valorar las fases de desarrollo de un programa: Análisis, diseño, implementación y prueba.
- Identificar las variables necesarias para desarrollar programas sencillos de cálculos matemáticos y de gestión de cadenas. Se han visto los tipos de datos básicos no estructurados.
- Los alumnos se han introducido en los operadores lógicos y nociones básicas de álgebra de Boole.
- Han desarrollado programas con estructuras sencillas de control de flujo.

4. EXPERIMENTO: PROTOTIPO PARA LA INVESTIGACIÓN. (ESTUDIO PREVIO)

4.5 Análisis estadístico

En la tabla 4.5 se relacionan las variables que se consideran en el estudio. Por una parte tenemos los datos académicos de los alumnos (notas medias de la 2ª y 3ª evaluación) y por otra parte los resultados de los tests pre y post formación en PC.

Grupo	Nº Alumnos	Pre-tests	Nota Media 2ªEvaluación	Nota Media 3ª Evaluación	Post-tests
Experimental	NE	PRETE	NM2EE	NM3EE	POSTE
Control	NC	PRETC	NM2EC	NM3EC	POSTC

Table 4.5: Variables del estudio

Para el tratamiento estadístico de los datos se ha utilizado el programa IBM SPSS Statistics Versión 22. En la tabla 4.6 se relacionan las variables que se han definido y en el Apéndice .2 se detallan los datos incorporados.

Nombre	Tipo	Etiqueta	Medida
NUM	Numérico	NÚMERO	Ordinal
SEXO	Numérico	SEXO	Nominal
MODALIDAD	Numérico	MODALIDAD	Nominal
GRUPO	Numérico	GRUPO	Nominal
NM2E	Numérico	Nota media 2ªEva.	Escala
NM3E	Numérico	Nota media 3ªEva.	Escala
PRET	Numérico	PRE-TEST	Escala
POST	Numérico	POS-TEST	Escala

Table 4.6: Variables definidas en SPSS

4.5.1 Estudio estadístico de la población

A la vista de los datos académicos de los alumnos (notas medias de la 2ª evaluación) y las puntuaciones obtenidas en los tests previos a la formación en PC hemos de verificar que los dos grupos son adecuados para realizar el experimento. Para comprobar que ambos grupos experimental y de control son dos muestras similares realizaremos las pruebas estadísticas para corroborar que ambas siguen una distribución normal y sus estadísticos son similares.

4.5 Análisis estadístico

En la tabla 4.7 se presentan algunos estadísticos descriptivos de las muestras de los resultados obtenidos en la nota media de la 2ª evaluación.

		EXPERIMENTAL		CONTROL	
		Estadístico	Error estándar	Estadístico	Error estándar
Media		5,0214	0,45297	4,6846	0,59127
95% de intervalo de confianza para la media	Límite inf.	4,0428		3,3964	
	Límite sup.	6,0000		5,9729	
Media recortada al 5%		5,0516		4,5551	
Mediana		5,0000		5,0000	
Varianza		2,873		4,545	
Desviación estándar		1,69487		2,13184	
Mínimo		2,00		1,70	
Máximo		7,50		10,00	
Rango		5,50		8,30	
Rango intercuartil		2,90		2,15	
Asimetría		-0,092	0,597	1,080	0,161
Curtosis		-1,057	1,154	2,434	1,191

Table 4.7: Estadísticos descriptivos de la variable “NM2E: Nota media 2ª evaluación”

Definiremos la hipótesis alterna H_1 : *El promedio de las notas medias de la 2ª evaluación de ambos grupos, experimental y de control, es distinto.*

La hipótesis nula H_0 se definirá: *No existe diferencia significativa entre las notas medias de la 2ª evaluación entre los dos grupos.*

El nivel de confianza de la prueba lo situaremos en el 95%. Determinamos $\alpha = 5\%$. Es un nivel generalmente utilizado en las ciencias sociales.

Para la elección de la prueba estadística consideraremos que estamos estudiando dos grupos el experimental y el de control. En este momento no debería haber diferencia entre las dos poblaciones. Nos encontramos con un estudio transversal pues estamos estudiando dos grupos en un mismo momento. También sabemos que la variable aleatoria NM2E es una variable numérica. Tenemos que utilizar una prueba t-student para muestras independientes.

4. EXPERIMENTO: PROTOTIPO PARA LA INVESTIGACIÓN. (ESTUDIO PREVIO)

La prueba t-student requiere de dos supuestos:

- Normalidad: La variable aleatoria se distribuye en ambos grupos normalmente. Dado que nos encontramos con una muestra pequeña (< 30 individuos) realizaremos la prueba de Shapiro Wilk.
- Igualdad de varianza: Realizaremos la prueba de Leneve para corroborarlo.

En la tabla 4.8 podemos comprobar con un nivel de confianza del 95% que ambas muestras se comportan siguiendo una distribución normal. Los niveles de significación para ambas muestras son ($0,563 > \alpha = 0,05$) y ($0,196 > \alpha = 0,05$)

GRUPO	Estadístico	gl	Sig.
EXPERIMENTAL	0,950	14	0,563
CONTROL	0,912	13	0,196

Table 4.8: Prueba de Normalidad Shapiro Wilk (Var.NM2E)

Respecto a la prueba de igualdad de varianzas, realizamos la prueba de Leneve 4.9. Podemos asegurar que ambas varianzas son iguales, con un nivel de significación ($0,875 > \alpha = 0,05$).

Se asumen varianzas iguales		
Prueba de Leneve de	F	0,025
calidad de varianzas	Sig.	0,875

Table 4.9: Prueba Leneve (Var. “NM2E: Nota media 2^a evaluación”)

Comprobados los dos supuestos para las muestras se ha realizado una comparación de medias mediante una prueba t-student para muestras independientes. Dado que en la prueba Leneve hemos asegurado que las varianzas son iguales podemos afirmar que el valor de significación para esta prueba es ($0,652 > \alpha = 0,05$). Los resultados obtenidos se detallan en la tabla 4.10.

Por lo tanto se acepta la hipótesis nula H_0 , es decir; no existe diferencia significativa entre las notas medias de la 2^a evaluación entre los dos grupos.

4.5 Análisis estadístico

		Se asumen varianzas iguales	No se asumen varianzas iguales
Prueba t para la igualdad de medias	t	0,456	0,452
	gl	25	22,929
	Sig. (bilateral)	0,652	0,655
	Diferencia de medias	0,33681	0,33681
	Diferencia de error estándar	0,73839	0,74484
	95% de intervalo de confianza de la diferencia	-1,18394 1,85756	-1,20426 1,87789

Table 4.10: Prueba t de la variable “NM2E: Nota media 2ª evaluación”

En la tabla 4.11 se presentan algunos estadísticos descriptivos de las muestras de los resultado obtenidos en los PRE-TESTS.

Del mismo modo que en el proceso anterior comprobamos que los valores obtenidos en los pre-tests en los dos grupos siguen una distribución normal y sus varianzas son iguales. Además al realizar la comparación de medias mediante la prueba t-student para muestras independientes concluimos que no existe diferencia significativa entre las notas PRE-TEST entre los dos grupos. Dado que en la prueba Leneve hemos asegurado que las varianzas son iguales podemos afirmar que el valor de significación para esta prueba es ($0,186 > \alpha = 0,05$). Los resultados obtenidos se detallan en la tabla 4.12.

Como conclusión del estudio estadístico de la población podemos afirmar que los dos grupos, experimental y de control son válidos para la prueba, dado que ambos siguen una distribución normal y sus estadísticos son similares. Estamos ante dos muestras similares independientes.

**4. EXPERIMENTO: PROTOTIPO PARA LA INVESTIGACIÓN.
(ESTUDIO PREVIO)**

		EXPERIMENTAL		CONTROL	
		Estadístico	Error estándar	Estadístico	Error estándar
Media		7,3143	0,35190	6,6154	0,31477
95% de intervalo de confianza para la media	Límite inf.	6,5541		5,9296	
	Límite sup.	8,0745		7,3012	
Media recortada al 5%		7,2825		6,5060	
Mediana		7,0000		6,4000	
Varianza		1,734		1,288	
Desviación estándar		1,31667		1,13493	
Mínimo		5,60		5,60	
Máximo		9,60		9,60	
Rango		4,00		4,00	
Rango intercuartil		2,50		1,25	
Asimetría		0,309	0,597	1,746	0,616
Curtosis		-1,435	1,154	3,255	1,191

Table 4.11: Estadísticos descriptivos de la variable “PRET: Resultados PRE-TEST”

		Se asumen varianzas iguales	No se asumen varianzas iguales
Prueba de Leneve de calidad de varianzas	F Sig.	1,848 0,186	
Prueba t para la igualdad de medias	t	1,472	1,480
	gl	25	24,874
	Sig. (bilateral)	0,154	0,151
	Diferencia de medias	0,69890	0,69890
	Diferencia de error estándar	0,47483	0,47214
	95% de intervalo de confianza de la diferencia	-0,27902 1,67682	-0,27373 1,67153

Table 4.12: Prueba t de la variable “PRET: Resultados PRE-TEST”

4.5.2 Resultados del experimento

Transcurrido el periodo de formación se ha procedido a recopilar los datos correspondientes a la tercera evaluación. Los datos obtenidos se muestran en el Apéndice .2 columna NM3E.

En la tabla 4.13 se presentan algunos estadísticos descriptivos de las muestras de los resultados obtenidos en la nota media de la 3ª evaluación.

		EXPERIMENTAL		CONTROL	
		Estadístico	Error estándar	Estadístico	Error estándar
Media		5,7214	0,60521	4,5308	0,70805
95% de intervalo de confianza para la media	Límite inf.	4,4139		2,9881	
	Límite sup.	7,0289		6,0735	
Media recortada al 5%		5,7627		4,4231	
Mediana		5,0000		4,0000	
Varianza		5,128		6,517	
Desviación estándar		2,26450		2,55290	
Mínimo		1,70		1,00	
Máximo		9,00		10,00	
Rango		7,30		9,00	
Rango intercuartil		3,70		2,50	
Asimetría		-0,019	0,597	1,124	0,616
Curtosis		-1,033	1,154	0,924	1,191

Table 4.13: Estadísticos descriptivos de la variable “NM3E: Nota media 3ª evaluación”

Definiremos la hipótesis alterna H_1 : *El promedio de las notas medias de la 3ª evaluación de ambos grupos, experimental y de control, es distinto.*

La hipótesis nula H_0 se definirá: *No existe diferencia significativa entre las notas medias de la 3ª evaluación entre los dos grupos.*

El nivel de confianza de la prueba los situaremos en el 95%. Determinamos $\alpha = 5\%$.

Para la elección de la prueba estadística realizaremos, al igual que con los datos de la 2ª evaluación, una prueba t-student para muestras independientes para aceptar o rechazar nuestra hipótesis.

4. EXPERIMENTO: PROTOTIPO PARA LA INVESTIGACIÓN. (ESTUDIO PREVIO)

La prueba t-student requiere de dos supuestos:

- Normalidad: La variable aleatoria se distribuye en ambos grupos normalmente. Dado que nos encontramos con una muestra pequeña (< 30 individuos) realizaremos la prueba de Shapiro Wilk.
- Igualdad de varianza: Realizaremos la prueba de Leneve para corroborarlo.

En la tabla 4.14 podemos comprobar con un nivel de confianza del 95% que ambas muestras se comportan siguiendo una distribución normal. Los niveles de significación para ambas muestras son ($0,215 > \alpha = 0,05$) y ($0,0,096 > \alpha = 0,05$)

GRUPO	Estadístico	gl	Sig.
EXPERIMENTAL	0,919	14	0,215
CONTROL	0,889	13	0,096

Table 4.14: Prueba de Normalidad Shapiro Wilk (Var.NM3E)

Respecto a la prueba de igualdad de varianzas, realizamos la prueba de Leneve 4.15. Podemos asegurar que ambas varianzas son iguales, con un nivel de significación ($0,978 > \alpha = 0,05$).

Se asumen varianzas iguales		
Prueba de Leneve de	F	0,001
calidad de varianzas	Sig.	0,978

Table 4.15: Prueba Leneve (Var. “NM3E: Nota media 3ª evaluación”)

Comprobados los dos supuestos para las muestras se ha realizado una comparación de medias mediante una prueba t-student para muestras independientes. Dado que en la prueba Leneve hemos asegurado que las varianzas son iguales podemos afirmar que el valor de significación para esta prueba es ($0,211 > \alpha = 0,05$). Los resultados obtenidos se detallan en la tabla 4.16.

Por lo tanto se acepta la hipótesis nula H_0 , es decir; no existe diferencia significativa entre las notas medias de la 3ª evaluación entre los dos grupos.

4.5 Análisis estadístico

		Se asumen varianzas iguales	No se asumen varianzas iguales
Prueba t para la igualdad de medias	t	1,284	1,278
	gl	25	24,077
	Sig. (bilateral)	0,211	0,213
	Diferencia de medias	1,19066	1,19066
	Diferencia de error estándar	0,92719	0,93146
	95% de intervalo de confianza de la diferencia	-0,71892 3,10024	-0,73145 3,11277

Table 4.16: Prueba t de la variable “NM3E: Nota media 3ª evaluación”

Aunque la prueba realizada nos indica que los resultados no son significativos podemos observar en la tabla 4.17 que las medias del grupo experimental han aumentado un porcentaje 13'94% mientras que las del grupo de control han disminuido un 3'28%. Esto podría indicar que la formación realizada ha influido positivamente sobre los resultados de los alumnos en el resto de las materias de ciencias.

	Nota media 2ª Evaluación		Nota media 3ª Evaluación	
	EXPERIMENTAL	CONTROL	EXPERIMENTAL	CONTROL
N	14	13	14	13
Media	5,0214	4,6846	5,7214	4,5308
Desv. estándar	1,69487	2,13184	2,26450	2,55290
Media de error estándar	0,45297	0,59127	0,60251	0,70805

Table 4.17: Comparación: “Notas medias 2ª y 3ª evaluación”

4. EXPERIMENTO: PROTOTIPO PARA LA INVESTIGACIÓN. (ESTUDIO PREVIO)

4.5.3 Limitaciones del estudio y recomendaciones

La duración de la formación ha sido de 18 horas, menor tiempo del esperado. El profesor esperaba dedicar más tiempo a esta formación pero se ha producido un retraso en los temas anteriores. Finalmente el POST-TEST no se ha realizado debido a cambios en la organización de las actividades de fin de curso. Se recomienda para futuros experimentos dejar más margen para ser capaces de adaptarse a cambios imprevistos.

La formación ha tenido como objetivo que los alumnos realizaran sus primeros pasos en programación. El objetivo debería haber sido que aprendieran las habilidades propuestas en el estudio, aquellas que definen el Pensamiento Computacional (análisis, abstracción y secuenciación). Para encontrar resultados significativos será necesaria mayor coordinación con del profesorado responsable de la formación.

Las diferencias obtenidas, no han sido significativas. La muestra ha sido pequeña. Para realizar este estudio con resultados más significativos será necesario ampliar la muestra a más centros con más grupos y de distintos niveles.

5

Conclusiones y trabajo futuro

5.1 Conclusiones

Este estudio tiene como objetivo aportar datos para poder evaluar la influencia del aprendizaje del Pensamiento Computacional en los resultados académicos de las asignaturas de ciencias en alumnos de secundaria.

El desarrollo del estudio comienza con una revisión de la literatura para definir el pensamiento computacional e identificar los principales métodos a la hora de afrontar la enseñanza del PC.

Para definir el pensamiento computacional hay diferentes puntos de vista. A partir de estos lo definimos como el siguiente conjunto de habilidades: **análisis, abstracción y secuenciación**. Estas habilidades van a permitir que los estudiantes sean capaces de adaptarse a los cambios que actualmente se están produciendo.

Los principales métodos para enseñar PC son:

- Enseñar Pensamiento Computacional *"sin programar"*
- Enseñar Pensamiento Computacional *"a través de la programación"*

Posteriormente se estudian algunas experiencias de enseñanza del PC de cada uno de estos métodos.

Para obtener datos contrastables se realiza un prototipo con un número limitado de alumnos. La realización de este estudio previo nos ha mostrado algunas de las dificultades que encontraremos cuando se realice a mayor escala. Según Weinberg (12)

5. CONCLUSIONES Y TRABAJO FUTURO

este tipo de experimentos es útil para la generación de hipótesis, pero nunca puede confirmar una hipótesis.

Aunque las pruebas realizadas nos indican que los resultados no son significativos se observa que el grupo experimental ha mejorado sus resultados académicos, en cambio, el grupo de control ha empeorado sus resultados. Esto podría indicar que la formación realizada ha influido positivamente sobre los resultados de los alumnos en el resto de las materias de ciencias. No obstante esta mejora podría deberse a otros factores no considerados. La existencia de un grupo de control puede minimizar este hecho puesto que los factores, en principio, afectarán de forma similar a los dos grupos.

5.2 Trabajo futuro

El experimento tendrá que realizarse con una muestra mayor para obtener resultados significativos que apoyen o no la hipótesis planteada por el estudio.

Con una muestra mayor se puede obtener información adicional al comparar el detalle de algunas variables como por ejemplo:

- En los tests de PC se puede diferenciar la puntuación de las tres habilidades definidas: análisis, abstracción y secuenciación. De esta forma podemos buscar correlaciones entre cada una de las habilidades estudiadas y la mejora de las medias académicas.
- Se pueden buscar correlaciones a partir de la variable sexo o el tipo de Bachillerato que cursan los alumnos, Bachillerato de Ciencias o de Ciencias Sociales y Humanidades.
- Conocer más detalles de las experiencias de enseñanza del pensamiento computacional en alumnos de secundaria clasificando la formación impartida durante el experimento, podría aportar correlaciones entre el tipo de formación y los resultados.

Apéndices

.1 Apéndice I

Test para el nivel 1º Bachiller

1. Comprueba si la ecuación representa cada una de las secuencias propuestas.
 - (a) Secuencia: 1, 2, 3, 4, 5, ...
ecuación: $n + 1$, donde valor de n se inicia desde 0.
 - (b) Secuencia: 2, 4, 6, 8, 10, ...
ecuación: $n + 2$, donde el valor de n se inicia desde 0.
 - (c) Secuencia: 5, 7, 9, 11, 13, ...
ecuación: $n + 2$, donde valor de n se inicia desde 5.
 - (d) Secuencia: 2, 5, 8, 11, 14, ...
ecuación: $3n-1$ donde el valor de n se inicia desde 1.
 - (e) Secuencia: 1, 3, 5, 7, 11, ...
ecuación: $2n-1$, donde valor de n se inicia desde 1.

2. ¿Cuáles son los siguientes números de las secuencias propuestas?
 - (a) 6, 10, 14, 18, 22, 26, 30, ..., ...
 - (b) 2, 44, 4, 41, 6, 38, 8, ..., ...
 - (c) 14, 14, 26, 26, 38, 38, 50, ..., ...
 - (d) 4, 7, 26, 10, 13, 20, ..., ...
 - (e) 13, 29, 15, 26, 17, 23, 19, ..., ...

3. ¿Qué tipo de datos (entero, carácter, tira de caracteres, número decimal) se necesita para almacenar la siguiente información?
 - (a) Nombre del estudiante
 - (b) Nota media de un estudiante
 - (c) Precio de un libro
 - (d) Número de teléfono
 - (e) Faltas de asistencia de un estudiante

-
4. ¿Cuáles son los datos de entrada y cuáles de salida en los siguientes problemas?
- (a) Calcular el área de un rectángulo cuya longitud es de 20 cm y la anchura es 30 cm.
 - (b) Calcular el promedio de 5 números a, b, c, d y e.
 - (c) Calcular la circunferencia de un círculo de radio 10 cm.
 - (d) Calcular cuál es el mayor de 10 números.
 - (e) Una terraza tiene una forma cuadrada de 12 metros de lado. ¿Cuál es el área del suelo de la terraza?
5. Señala de las siguientes series de números cuál sigue una ordenación ascendente:
- (a) 408, 399, 387, 376, 377, 367
 - (b) 354, 367, 377, 398, 403, 412
 - (c) 414, 401, 383, 374, 381, 379
 - (d) 423, 424, 391, 384, 375, 381
6. Señala de las siguientes series de números cuál sigue una ordenación descendente:
- (a) 225, 358, 468, 589, 698, 888
 - (b) 678, 669, 543, 365, 412, 221
 - (c) 396, 369, 354, 345, 321, 312
 - (d) 543, 534, 509, 469, 491, 419
7. ¿Cuál de las siguientes secuencias permite determinar el mayor de dos números?
- (a) Paso 1. Si $A > B$ entonces escribir “A es el mayor”
Paso 2. Leer dos números
Paso 3. De lo contrario escribir “B es el mayor”
Paso 4. Si $A=B$ volver a leer dos números
 - (b) Paso 1. Leer dos números
Paso 2. Si $A > B$ entonces escribir “A es el mayor”
Paso 3. De lo contrario escribir “B es el mayor”
Paso 4. Si $A=B$ volver a leer dos números

(c) Paso 1. Leer dos números

Paso 2. Si $A=B$ volver a leer dos números

Paso 3. Si $A > B$ entonces escribir “A es el mayor”

Paso 4. De lo contrario escribir “B es el mayor”

.2 Apéndice II

DATOS 01.sav

	NUM	SEXO	MODALIDAD	GRUPO	NM2E	NM3E	PRET
1	1	FEMENINO	BACH	DESCARTADOS	.	.	.
2	2	FEMENINO	BACH	CONTROL	4,00	4,00	5,70
3	3	MASCULINO	BAC	CONTROL	5,00	3,00	6,00
4	4	MASCULINO	BAC	EXPERIMENTAL	5,30	5,70	8,80
5	5	MASCULINO	BAC	EXPERIMENTAL	3,50	4,50	6,80
6	6	MASCULINO	BAC	CONTROL	5,00	5,00	6,40
7	7	FEMENINO	BAC	EXPERIMENTAL	4,70	4,70	6,40
8	8	MASCULINO	BACH	CONTROL	5,00	5,00	7,00
9	9	FEMENINO	BAC	EXPERIMENTAL	4,00	4,00	6,40
10	10	FEMENINO	BACH	DESCARTADOS	.	.	.
11	11	MASCULINO	BAC	EXPERIMENTAL	7,00	8,50	6,00
12	12	MASCULINO	BAC	EXPERIMENTAL	2,00	1,70	5,60
13	13	MASCULINO	BACH	DESCARTADOS	.	.	.
14	14	MASCULINO	BACH	EXPERIMENTAL	4,00	5,00	7,20
15	15	FEMENINO	BACH	DESCARTADOS	.	.	.
16	16	MASCULINO	BAC	EXPERIMENTAL	6,70	8,00	8,80
17	17	MASCULINO	BAC	CONTROL	6,50	9,00	7,20
18	18	MASCULINO	BACH	DESCARTADOS	.	.	.
19	19	MASCULINO	BACH	DESCARTADOS	.	.	.
20	20	FEMENINO	BAC	EXPERIMENTAL	5,30	4,70	6,00
21	21	FEMENINO	BACH	CONTROL	5,00	6,00	6,00
22	22	MASCULINO	BACH	CONTROL	3,00	3,00	6,00
23	23	FEMENINO	BACH	CONTROL	4,00	3,30	5,60
24	24	FEMENINO	BACH	DESCARTADOS	.	.	.
25	25	MASCULINO	BAC	EXPERIMENTAL	7,50	9,00	8,00
26	26	MASCULINO	BAC	EXPERIMENTAL	7,00	8,00	9,60
27	27	FEMENINO	BACH	CONTROL	6,00	4,00	6,50
28	28	FEMENINO	BACH	DESCARTADOS	.	.	.
29	29	MASCULINO	BAC	CONTROL	2,00	2,30	5,60
30	30	MASCULINO	BAC	CONTROL	1,70	1,00	6,40
31	31	MASCULINO	BAC	EXPERIMENTAL	6,30	8,30	8,40
32	32	MASCULINO	BACH	CONTROL	10,00	10,00	9,60
33	33	MASCULINO	BACH	EXPERIMENTAL	4,00	5,00	6,00
34	34	FEMENINO	BACH	DESCARTADOS	.	.	.
35	35	MASCULINO	BAC	EXPERIMENTAL	3,00	3,00	8,40
36	36	MASCULINO	BAC	CONTROL	3,70	3,30	8,00

17/06/16 22:07

1/2

Figure 1: Conjunto de datos tratados con SPSS

Referencias

- [1] ALAN BUNDY. **Computational Thinking is Pervasive.** *Thinking*, pages 1–3, 2007. 1
- [2] PETER B HENDERSON. **Ubiquitous computational thinking.** *Computer*, **42**(10):100–102, 2009. 1
- [3] JEANNETTE M WING. **Computational thinking.** *Communications of the ACM*, **49**(3):33–35, 2006. 1, 9
- [4] JEANNETTE M. WING. **Computational Thinking: What and Why?** *thelink - The Magazine of the Varnege Mellon University School of Computer Science*, **1**(March 2006):1–6, 2011. 1, 9, 11, 23
- [5] S. GROVER AND R. PEA. **Computational Thinking in K-12: A Review of the State of the Field.** *Educational Researcher*, **42**(1):38–43, 2013. 1
- [6] CONSELLERIA D'EDUCACIÓ FORMACIÓ I OCUPACIÓ. **DECRETO 112/2007, de 20 de julio, del Consell, por el que se establece el currículo de la Educación Secundaria Obligatoria en la Comunitat Valenciana. [2007/9717].** *Dogv*, **5562**:186, 2007. 3
- [7] CONSELLERIA D'EDUCACIÓ FORMACIÓ I OCUPACIÓ. **DECRETO 87/2015, de 5 de junio, del Consell, por el que establece el currículo y desarrolla la ordenación general de la Educación Secundaria Obligatoria y del Bachillerato en la Comunitat Valenciana. [2015/5410].** *Dogv*, **7544**:16325–16694, 2014. 3
- [8] KYU HAN KOH. *Computational Thinking Pattern Analysis: a Phenomenological Approach To Compute.* Phd thesis, University of Colorado, 2014. 4, 23
- [9] JUAN CARLOS OLABE, XABIER BASOGAIN, MIGUEL ÁNGEL OLABE, INMACULADA MAÍZ, AND CARLOS CASTAÑO. **Solving math and science problems in the real world with a computational mind.** *Journal of New Approaches in Educational Research*, **3**(2):72–78, 2014. 4, 23
- [10] DAVID HEMMENDINGER. **A Plea for Modesty.** *ACM Inroads*, **1**(2):4–7, 2010. 9
- [11] A. V. AHO. **Computation and Computational Thinking.** *The Computer Journal*, **55**(7):832–835, 2012. 9
- [12] ANDREA ELIZABETH WEINBERG. *Computational Thinking: An Investigation of the existing scholarship and research.* Phd thesis, Colorado State University, 2013. 10, 37
- [13] COMPUTER SCIENCE TEACHERS ASSOCIATION (CSTA), INTERNATIONAL SOCIETY FOR TECHNOLOGY EDUCATION (ISTE), AND NATIONAL SCIENCE FOUNDATION (NSF). **Operational Definition of Computational Thinking for K–12 Education.** *CSTA*, page 1030054, 2011. 10
- [14] ELAINE KAO. **Exploring Computational Thinking,** 2010. 10
- [15] MARK GUZDIAL. **Education Paving the way for computational thinking.** *Communications of the ACM*, **51**(8):25, 2008. 12
- [16] JAMES J LU AND GEORGE H L FLETCHER. **Thinking About Computational Thinking Categories and Subject Descriptors.** *Sigcse*, pages 260–264, 2009. 12
- [17] MIT. **Scratch.** 13, 16
- [18] MIT. **App Inventor.** 13
- [19] CATHERINE WILSON GILLESPIE. **La visión de Seymour Papert para la educación de la niñez.** *ECRP - Investigación práctica de la niñez temprana*, **6**(1), 2004. 13
- [20] CS EDUCATION RESEARCH GROUP. **CS Unplugged.** 15
- [21] GLORIA CECILIA RIOS MUÑOZ. *SCRATCH + ABP, Como estrategia para el desarrollo del Pensamiento Computacional.* Phd thesis, UNIVERSIDAD EAFIT (Medellín - Colombia), 2015. 16
- [22] XABIER BASOGAIN-OLABE, MIGUEL ÁNGEL OLABE-BASOGAIN, AND JUAN CARLOS OLABE-BASOGAIN. **Pensamiento Computacional a través de la Programación: Paradigma de Aprendizaje.** *Revista de Educación a Distancia (RED)*, **1**(46):33, sep 2015. 17
- [23] R MORELLI, T DE LANEROLLE, P LAKE, N LIMARDO, B TAMOTSU, AND C UCHE. **Can Android App Inventor Bring Computational Thinking to K-12?** *Proc. 42nd ACM technical symposium on Computer science education (SIGCSE'11)*, 2011. 18
- [24] I MONTERO AND OG LEÓN. *Métodos de Investigación en psicología y Educación.* Madrid: Mc-Graw-Hill, 2007. 22
- [25] BRIONY J OATES. *Researching information systems and computing.* SAGE Publications, 2006. 22
- [26] CARLOS ARMANDO PINILLA, ALEJANDRO MORENO, LOPEZ SOLANO. **Gambas ! Mucho más que un plato de mariscos.** *Revista Inventum*, **1**(4):55–60, 2008. 26
- [27] GAMBAS ES.ORG. **Comunidad gambas-es.** 27

Declaración

Por la presente declaro que he producido este documento sin la ayuda de terceros y sin hacer uso de ayudas distintas de las especificadas; ideas tomadas en forma directa o indirecta de otras fuentes se han identificado como tal. Este documento no ha sido previamente presentado de manera idéntica o similar a cualquier otro tribunal español o extranjero. El trabajo de tesis se realizó durante el curso 2015/2016.

Valencia, 29 de junio de 2016

FITXA DEL TREBALL FINAL

Títol del treball:	Estudi de la influència de l'aprenentatge del pensament computacional en les matèries de ciències en alumnes de secundària
Nom de l'autor:	<i>Juan José Boix Tormos</i>
Nom del consultor/a:	<i>Enrique Guaus i Termens</i>
Nom del PRA:	<i>David García Solórzano</i>
Data de lliurament (mm/aaaa):	<i>06/2016</i>
Titulació o programa:	<i>Màster Universitari d'Aplicacions Multimèdia</i>
Àrea del Treball Final:	<i>Treball de Final de Màster de Recerca</i>
Idioma del treball:	<i>Castellà</i>
Paraules clau	<i>Pensament computacional, secundària, ensenyament de les ciències</i>
Resum del Treball (màxim 250 paraules):	
<p>El pensament computacional és una de les estratègies d'aprenentatge que destaquen en l'actualitat que té com a objectiu preparar els estudiants a les demandes de la societat actual. Aquest estudi aporta una definició del Pensament Computacional com el següent conjunt d'habilitats: anàlisi, abstracció i seqüenciació. També aporta dades per verificar si el pensament computacional és útil, no només directament en les ciències de la computació; sinó que, el seu coneixement aporta beneficis a la resta de matèries de ciències. Per això, aquest article presenta un experiment en què es parteix de dos grups d'alumnes de secundària. A un d'ells se li forma en PC durant un període de temps i l'altre no. Transcorreguda la formació s'avalua amb proves estadístiques les diferències en els resultats acadèmics de les assignatures de ciències, d'aquells alumnes de secundària que l'han estudiat a diferència de l'altre grup que actua com a grup de control.</p>	
Abstract (in English, 250 words or less):	
<p>Computational thinking is one of the learning strategies that emphasize today that aims to prepare students for the demands of today's society. This study provides a definition of Computational thinking as the next set of skills: analysis, abstraction and sequencing. It also provides data to verify if the Computational Thinking is useful, not only directly in computer science; but that knowledge brings benefits to other science subjects. To this end, this paper presents an experiment that begins with two groups of high school students. One of them are trained in Computational Thinking for a period of time and the other not. Elapsed training is evaluated with statistical tests differences in academic outcomes of science subjects, those high school students who have studied unlike the other group that acts as a control group.</p>	