

Susana Masapanta Carrión
J. Ángel Velázquez Iturbide

**Una Revisión Sistemática del Uso
de la Taxonomía de Bloom en la
Enseñanza de la Informática**
Número 2017-02

Serie de Informes Técnicos DLSI1-URJC
ISSN 1988-8074
Grupo Docente de Lenguajes y Sistemas Informáticos I
Universidad Rey Juan Carlos

Índice

1	Introducción	1
2	Metodología de la Investigación	2
2.1	Fuentes de Información.....	2
2.2	Términos de Búsqueda.....	3
2.3	Criterios de Selección	3
2.4	Preguntas de Investigación	4
2.5	Metodología de Análisis	4
3	Resultados	5
3.1	Resultados preguntas sin categorización	5
3.2	Resultados de las preguntas con categorización	6
3.2.1	Materias donde se utilizó la taxonomía de Bloom	6
3.2.2	Usos de la taxonomía de Bloom.....	7
3.2.3	Dificultades en el uso de la taxonomía de Bloom	8
4	Resumen de Hallazgos	17
5	Discusión.....	18
6	Conclusiones	19
	Agradecimientos	19
	Referencias.....	19
	Apéndice A: Lista de los artículos seleccionados para la revisión sistemática.	21

Una Revisión Sistemática del Uso de la Taxonomía de Bloom en la Enseñanza de la Informática

Susana Masapanta Carrión¹; J. Ángel Velázquez Iturbide²

¹ Facultad de Ingeniería, Pontificia Universidad Católica del Ecuador, Av. 12 de Octubre 1076 y Roca, 17012184, Quito, Ecuador

² Escuela Técnica Superior de Ingeniería Informática, Universidad Rey Juan Carlos, C/ Tulipán s/n, 28933, Móstoles, España

¹ smmasapanta@puce.edu.ec; ² angel.velazquez@urjc.es

Resumen. El uso de la taxonomía de Bloom ha sido considerado en la enseñanza de diferentes áreas del conocimiento y ha dado buenos resultados, sin embargo, en la enseñanza de la Informática se observa que existen complicaciones en su aplicación. La presente revisión sistemática presenta los resultados del análisis realizado para determinar las principales dificultades que tuvieron los autores en el uso de la taxonomía de Bloom, así como las posibles causas de esas dificultades y soluciones dadas a las mismas.

Palabras clave: Taxonomía de Bloom, enseñanza de la Informática, enseñanza de la programación.

1 Introducción

La taxonomía de Bloom fue creada como un marco de referencia para categorizar lo que se espera que los alumnos aprendan de la educación. La categorización se logra con el análisis del proceso cognitivo del alumno usando seis niveles taxonómicos que van de los más sencillos a lo más complejo y donde cada nivel depende de los conocimientos adquiridos en los niveles anteriores. La versión original [1] fue publicada en 1956 y la versión revisada [2] en 2001.

La taxonomía de Bloom ha sido considerada un referente para el diseño de pruebas y el desarrollo curricular en todo el mundo [2]. El currículo ACM / IEEE de Informática [3] especifica los objetivos de aprendizaje sobre la base de la versión revisada de la taxonomía de Bloom [4]. A pesar de las ventajas del uso de la taxonomía surgen dificultades al aplicarla, en especial en la enseñanza de la programación [5].

En el trabajo realizado por Thompson et al. [5] un grupo de cinco investigadores utilizó la versión revisada de la taxonomía de Bloom para categorizar las preguntas de los exámenes de programación de primer año. Los exámenes provenían de seis instituciones de Australasia y Estados Unidos. Terminada la fase de categorización, los investigadores habían clasificado una misma pregunta en distintos niveles de la

taxonomía. Thompson et al. [5] distinguen tres motivos para esta variedad de resultados: el primero es la dificultad para comprender las tareas cognitivas descritas en la taxonomía en el dominio de la programación, el segundo es la falta de ejemplos en programación y el tercero es la implicación del investigador en el contexto de enseñanza del curso en donde se aplicó la evaluación.

Un caso similar sucede en el trabajo realizado por Johnson et al. [6]. Los investigadores encontraron difícil llegar a un acuerdo sobre la clasificación de las evaluaciones usando la versión original de la taxonomía de Bloom. Johnson et al. [6] consideran que el uso de taxonomías no es fácil y en su artículo describen algunos de los estudios realizados para determinar si la taxonomía es apropiada para la Informática.

Por otra parte, Fuller et al. [7] revisa la literatura sobre diferentes taxonomías educativas y su uso en la Informática. Identifica algunos problemas que surgen de su uso y propone una nueva taxonomía que pueda ser utilizada en cursos de programación. Este trabajo nos pareció interesante ya que los autores indican que la taxonomía de Bloom es la más usada. Si bien determinan las dificultades de uso, se diferencia de nuestro trabajo porque nuestro objetivo es recopilar y estructurar la información disponible sobre las dificultades que comentan los educadores en Informática, así como las causas y las soluciones a esas dificultades.

En este informe se presenta una revisión sistemática del uso de la taxonomía de Bloom en la enseñanza de la Informática. La estructura del informe es la siguiente: en el primer apartado se explica la metodología utilizada para realizar la revisión, en el segundo se presentan los resultados obtenidos de la búsqueda en las fuentes de información y las respuestas obtenidas a las preguntas de investigación, y en el apartado tercero se exponen las conclusiones a las que llegamos.

2 Metodología de la Investigación

En esta sección se detalla el proceso que se siguió para la revisión sistemática, incluye las fuentes de información, los términos de búsqueda, los criterios de inclusión y las preguntas de investigación. Se tomó como guía algunas de las pautas utilizadas por Britto y Usman [8] para realizar este trabajo.

2.1 Fuentes de Información

Se seleccionó las revistas y los congresos más relevantes en la enseñanza de la Informática, la lista de fuentes se especifica a continuación:

- Revista Computer Science Education.
- Revista Transactions on Computing Education (TOCE).
- Congreso International Computing Education Research (ICER).
- Congreso Special Interest Group on Computer Science Education (SIGCSE).
- Congreso Innovation and Technology in Computer Science Education (ITiCSE).
- Congreso Australasian Computing Education Conference (ACE).

- Congreso International Conference on Computing Education Research (Koli Calling).

2.2 Términos de Búsqueda

Para identificar las palabras claves se consideró el uso de la taxonomía de Bloom en la Informática y en programación, también se han considerado las distintas formas de referirse a la taxonomía de Bloom. La cadena de búsqueda utilizada fue:

```
("bloom's taxonomy" OR "bloom taxonomy" OR "cognitive taxonomy")
AND
(programming OR "computer science")
```

Para la revista Computer Science Education se buscó en la versión online de Tylor & Francis y para el resto de revistas y congresos se buscó dentro de la biblioteca digital de ACM.

2.3 Criterios de Selección

La búsqueda con la cadena presentada anteriormente nos arrojó 314 artículos, sin embargo, no todos ellos nos eran útiles. Se incluyeron criterios de exclusión e inclusión para determinar los artículos más relevantes.

El proceso de selección tuvo tres etapas, en la primera se eliminó los artículos duplicados y las publicaciones que correspondían a un volumen de actas de congreso, pero que no contenían ninguna comunicación del mismo. Quedando 306 artículos.

En la segunda se aplicaron los siguientes criterios de inclusión: a) Que en el resumen del artículo haga referencia al uso de la taxonomía de Bloom o b) Que en las palabras clave se encuentre "taxonomía de Bloom" o "Bloom". Esto ayudo a reducir el número de artículos a 45.

Finalmente, en la tercera etapa hubo un segundo proceso de exclusión debido a que, al comenzar con el análisis de los 45 artículos se observó que algunos trabajos provenían de una misma investigación, sólo que quizá presentado de forma distinta (p.ej. uno es más largo y otro más resumido). Por esta razón se decidió conservar el artículo que brinde más detalles para esta investigación. A continuación, se muestran los artículos que provienen de un mismo trabajo y qué artículo se seleccionó.

- (1) Autor: Satu Alaoutinen
Artículos similares - Códigos: 18, 37, 41
Queda el 41
- (2) Autores: Raymond Lister, John Leaney
Artículos similares - Códigos: 5, 29
Queda el 5
- (3) Autores: Dave Oliver, Tony Dobebe, Myles Greber, Tim Roberts
Artículos similares - Códigos: 6, 39
Queda el 39

- (4) Autores: Orni Meerbaum-Salant Michal Armoni Mordechai (Moti) Ben-Ari
 Artículos similares - Códigos: 10, 43
 Queda el 10

Luego de esto se codifico nuevamente los artículos para su identificación. La Tabla 1 resume el proceso descrito de selección, con las cifras de publicaciones resultantes de cada paso.

Tabla 1. Resultados búsqueda y aplicación de criterios de selección.

Fuente de información	Artículos encontrados	Artículos sin duplicados	Artículos Satisfacen criterios de inclusión	Artículos sin similares
Computer Science Education	7	7	7	6
ICER	42	39	5	5
TOCE	17	17	1	1
SIGCSE	119	117	13	11
ITiCSE	86	85	11	10
ACE	30	29	7	6
Koli Calling	13	12	1	1
Total	314	306	45	40

2.4 Preguntas de Investigación

Las siguientes preguntas de investigación nos permitieron tener una visión amplia del uso de la taxonomía de Bloom:

1. ¿Qué taxonomía de Bloom utilizan?
2. ¿Utilizan alguna otra taxonomía?
3. ¿En qué materias utilizan la taxonomía de Bloom?
4. ¿Para qué utilizan la taxonomía de Bloom?
5. ¿Comentan alguna dificultad de uso de la taxonomía de Bloom? En caso afirmativo indicar que dificultades encontraron.

2.5 Metodología de Análisis

La pregunta 1 sólo admitía dos respuestas (taxonomía original o revisada), por lo que simplemente se contaron las respuestas obtenidas. Algo similar sucede con la pregunta 2. Sin embargo, las preguntas 3-5 admitían una gran variedad de respuestas, por lo que se realizó un análisis cualitativo de las mismas.

Se elaboró una matriz para registrar las respuestas obtenidas de los 40 artículos seleccionados. Al analizar cada artículo, solamente se tuvieron en cuenta los comentarios propios de sus autores, no las citas a comentarios de otros investigadores.

El análisis cualitativo se realizó sin partir de categorías preestablecidas, como preconiza la *grounded theory* [9], y a través de numerosas iteraciones. Sin entrar en detalle de todas las iteraciones realizadas, podemos agruparlas en dos etapas:

1. Las dificultades se dividieron en dificultades internas de la taxonomía (inherentes a la taxonomía) y externas (relacionadas con quién usa la taxonomía). A su vez, cada categoría incluía diversas subcategorías. Sin embargo, esta forma de analizar no era un instrumento claro de clasificación de los comentarios, con frecuente incertidumbre sobre la subcategoría donde mejor encajaba una dificultad. Finalmente, se descartó esta clasificación.
2. Se distinguió si cada comentario era una expresión de una dificultad, una causa de la misma o una solución adoptada por los autores. Esta clasificación fue más puntual y factible de realizar, por lo que se eligió como válida.

De todas formas, cada etapa incluyó diversas iteraciones, con frecuentes idas y venidas entre los artículos. La mayor parte de las iteraciones de análisis fue realizada por la primera autora, aunque el segundo autor también realizó algunas. El proceso de análisis terminó tras llegar a un consenso.

3 Resultados

Los resultados están divididos en dos partes: resultados de las preguntas sin categorización y resultados de las respuestas para las cuales se crearon categorías.

3.1 Resultados preguntas sin categorización

La taxonomía original fue utilizada en el 75% de los artículos (véase Tabla 2), mientras que el 25% restante utilizó la taxonomía revisada.

Tabla 2. Taxonomía que utilizaron.

Taxonomía utilizada	Núm. de artículos	Porcentaje
Versión original de la taxonomía de Bloom (1956)	30	75%
Versión revisada de la taxonomía de Bloom (2001)	10	25%
Total	40	100%

En cuanto al uso de otras taxonomías, la mayor parte de los artículos usaron exclusivamente la taxonomía de Bloom (85%). Un 10% usaron la taxonomía SOLO [10] tras descartar el uso de Bloom. Finalmente, el 5% restante usaron una combinación de las taxonomías de Bloom y SOLO.

Tabla 3. Uso de otra taxonomía.

Usa otra taxonomía	Núm. de artículos	Porcentaje
No	34	85%
Taxonomía de SOLO	4	10%
Combinación de las taxonomías de Bloom y SOLO	2	5%
Total	40	100%

3.2 Resultados de las preguntas con categorización

El total de respuestas a las preguntas con categorización varía con respecto al número de los artículos seleccionados (40). Ya que en algunos de los artículos encontramos más de una respuesta e incluso hubo casos en los que no encontramos respuestas. Por ejemplo, en un mismo artículo encontramos una o más causas de las dificultades señaladas por los autores al usar Bloom, pero ninguna solución.

3.2.1 Materias donde se utilizó la taxonomía de Bloom

Al analizar en qué materias fue utilizada la taxonomía de Bloom se observó que además del uso en materias de pregrado, se la utilizó en un nivel de estudio en particular como primaria o secundaria. También se notó que en dos artículos se aplica en tres materias y en un artículo se aplicó en dos, con lo cual el número de materias contabilizadas llega a 43. Los resultados se muestran en la Tabla 4.

Tabla 4. Materias donde se utilizó la taxonomía de Bloom.

Materia	Núm. respuestas	Porcentaje
Programación en primer año	23	53,5%
Estructuras de datos y algoritmos	4	9,3%
Otras asignaturas de programación	3	7,0%
Ingeniería del software	5	11,6%
Otras materias de informática	4	9,3%
Nivel preuniversitario	4	9,3%
TOTAL	43	100%

En la categoría **Programación en primer año** se agruparon las materias de programación dictadas en primer año de Informática (CS1, Introducción a la programación procedimental u Orienta a Objetos) y si se desglosa por el paradigma de programación utilizado, obtenemos los resultados de la Tabla 5.

En la categoría **Estructuras de datos y algoritmos** encontramos tres artículos que hacen referencia a la materia de Estructuras de Datos y Algoritmos y uno a los contenidos de Tipos de datos abstractos y Recursividad.

En **Otras asignaturas de programación** se categorizó aquellas materias que tienen un nivel de programación más avanzado como Lenguajes de programación y Programación funcional.

Para **Ingeniería del software** se agrupó cuatros artículos relacionados con Ingeniería de Software en general y uno relacionado con Ingeniería de Requerimientos.

En **Otras materias de informática** incluye Bases de datos, Redes y Comunicaciones, Seguridad, Interacción Humano-Computador.

En la última categoría, **Nivel preuniversitario** se encuentran aquellas investigaciones realizadas en primaria o secundaria, aquí también se agrupó al

examen *AP Computer Science A Exam*. dado que se aplica a estudiantes de bachillerato.

Tabla 5. Paradigma de programación usado en primer año.

Paradigma de programación	Núm. Respuestas	Porcentaje
Programación procedimental	6	26,1%
Programación Orientada a Objetos	7	30,4%
Indeterminado	10	43,5%
Total	23	100%

3.2.2 Usos de la taxonomía de Bloom

Los usos que los autores dieron a la taxonomía de Bloom fueron diversos, por lo que se realizó un análisis para determinar las categorías de uso. Se realizaron cuatro iteraciones, en cada una de ellas se buscó reducir el número de categorías encontradas agrupando los usos por sus similitudes. Es así que de 18 categorías en la primera iteración se logró disminuir a 5 categorías en la cuarta iteración.

En esta última iteración se consideró indicar cuatro aspectos que ayudaron a definir las categorías. Tres de estos aspectos están relacionados con la docencia: definir objetivos de aprendizaje, actividades de instrucción y evaluación de los alumnos. Y el cuarto aspecto señala que no se sabe bien como se ha usado la taxonomía.

El análisis también permitió determinar que uno de los artículos informa de dos fines, lo que da un total de 41 usos (véase la Tabla 6).

Tabla 6. Usos de la taxonomía de Bloom

Categoría de uso	Subcategorías	Núm. respuestas	Total por categoría	Porcentaje
Evaluar a los alumnos	Desarrollar preguntas	10	19	46,3%
	Clasificar preguntas	7		
	Clasificar el aprendizaje	2		
Planificar la actividad docente	-	2	2	4,9%
Especificar objetivos de aprendizaje	-	2	2	4,9%
Otros	Crear una nueva taxonomía	2	3	7,3%
	Desarrollar un software educativo	1		
Indeterminado	-	15	15	36,6%
TOTAL		41	41	100%

A continuación, se describen las categorías de uso identificadas:

- Evaluar a los alumnos: la taxonomía se usa para medir los conocimientos del alumno en una asignatura. Esta categoría integra tres subcategorías:
 - a) Desarrollar preguntas o problemas según sus niveles cognitivos.
 - b) Clasificar preguntas o problemas ya desarrollados en niveles cognitivos.
 - c) Clasificar el aprendizaje de los alumnos. Se clasifica el rendimiento académico de los alumnos en niveles de la taxonomía de Bloom.
- Planificar la actividad docente (2 artículos). Se usa la taxonomía de Bloom para planificar la actividad docente de una asignatura de forma que mejore el aprendizaje de los alumnos.
- Especificar o clasificar los objetivos de aprendizaje de una asignatura (2 artículos).
- Otros:
 - a) Crear una nueva taxonomía (2 artículos). Se modifica para obtener a una nueva taxonomía, más adecuada para la informática.
 - b) Desarrollar software educativo (1 artículo). Se usa como base para el desarrollo de un tutorial para aprender a usar la propia taxonomía de Bloom.
- Indeterminado. Señalan el uso de la taxonomía de Bloom, pero sin dar más detalles. En algunos casos, sólo es mencionada.

3.2.3 Dificultades en el uso de la taxonomía de Bloom

El 37,5% de los artículos analizados indican que los investigadores tuvieron dificultades al utilizar la taxonomía de Bloom (Véase Tabla 7), mientras que un 62,5% no comentan haber tenido alguna dificultad

Tabla 7. Encuentran dificultades de uso de la taxonomía de Bloom.

Informes de dificultades de uso de la taxonomía de Bloom	Núm. artículos	Porcentaje
Si	15	37.5%
No	25	62.5%
Total	40	100%

Para el análisis que viene a continuación nos centramos en los 15 artículos que comentan haber tenido dificultades. Se presentan primero las dificultades, luego las causas y por último las soluciones que identificamos de los comentarios de los autores.

Para las dificultades se estableció cuatro categorías (véase Tabla 8). En tres artículos se encontraron dos dificultades con lo que el número de dificultades llega a 18.

Los resultados indican que “clasificar los objetivos o tareas de evaluación” dentro de los niveles de la taxonomía representa la mayor dificultad que experimentaron los autores (77,77%).

A continuación, se muestra la descripción de las categorías de dificultades:

- Aplicabilidad para clasificar los objetivos o las tareas de evaluación: presentan problemas al tratar de determinar en qué nivel de la taxonomía debería estar un objetivo, un contenido o una pregunta de evaluación de Informática. Un comentario que explica esta dificultad es el realizado por Whalley et al. [11] que dice: “el categorizar las preguntas de programación por la complejidad cognitiva aplicando la taxonomía de Bloom, ha demostrado ser un desafío incluso para un grupo experimentado de educadores de programación”.
- Especificar de forma precisa el conocimiento objeto de cada objetivo de aprendizaje o prueba de evaluación: El problema surge cuando el profesor va cambiando sin advertirlo entre conceptos relacionados entre los que hay ciertas diferencias. Por ejemplo, Starr *et al.* [12] advierten del cambio entre “iteración” y “bucle for”
- Aplicabilidad para medir el progreso del alumno: hay dificultades al ver si los procesos cognitivos de un estudiante usados para resolver un problema, progresan de forma ordenada siguiendo los niveles de la taxonomía. P.ej. Meerbaum-Salant et al. [13] comentan que “queríamos trabajar con una taxonomía estrictamente jerárquica que permitiera monitorear el progreso del estudiante, pero que coincidiera con el contexto del estudio y sus objetivos”.
- Comprensibilidad: se tienen dudas de cómo interpretar algunos de los términos de la taxonomía en un ambiente informático. Un ejemplo es el comentado por Thompson et al. [5] que explica que es difícil aclarar lo que significa aplicar un proceso y/o crear un proceso usando la taxonomía en programación.

Causas de las dificultades

Para las causas se elaboraron cinco categorías, algunas de ellas tienen subcategorías (Véase la Tabla 9). En algunos trabajos los autores manifiestan más de una causa para una misma dificultad (Véase la Tabla 11), por lo que el número total de causas es de 26.

Al observar los resultados podemos ver que no hay una categoría con la mayoría de resultados, pero se puede ver que hay cuatro causas principales: la necesidad de conocer el contexto educativo, la estructura de la taxonomía, la terminología de la taxonomía, y la comprensión de la taxonomía.

A continuación, se describe cada una de las categorías de causas:

- Necesidad de conocer el contexto educativo: los autores comentan que desconocer la forma como se enseñó a los alumnos el contenido evaluado, puede dar lugar a dificultades. Así p.ej. lo expresan Gluga et al. [4]. al destacar “la estrecha dependencia del conocimiento del contexto de la enseñanza para clasificar correctamente las preguntas del examen con Bloom”. Tiene dos subcategorías:

- a) Los alumnos utilizan distinto esfuerzo cognitivo para resolver un mismo ejercicio: los estudiantes pueden razonar de diferentes maneras para resolver un mismo ejercicio y estas pueden ser clasificadas en diferentes niveles.
- b) Conocer la forma de enseñar el contenido: la manera como se enseñe al estudiante influye en la manera cómo responderá en la evaluación y esta respuesta puede ser clasificada en un nivel diferente de la taxonomía del usado para instruir.
- Limitaciones de la taxonomía: corresponde a aquellas causas que son inherentes a la taxonomía como su definición o su estructura. Sus subcategorías son:
 - a) Conjunto incompleto o inadecuado de niveles para las tareas de programación. los niveles de la taxonomía, difícilmente se adaptan a los conceptos y tareas requeridas en programación [11]
 - b) Conjunto solapado: los niveles no están bien diferenciados entre ellos, lo que produce que una pregunta o un contenido pueda ser categorizado en dos niveles alternativos.
 - c) Concebida para evaluar y no para especificar objetivos: la taxonomía de Bloom fue creada con el fin de evaluar.
- Terminología: la terminología utilizada en informática, en especial aquella que se usa en programación, tiene diferente connotación de la utilizada en la taxonomía de Bloom. A su vez, podemos distinguir entre:
 - a) Terminología extraña para programación.
 - b) Falta de ejemplos.
- Comprensión deficiente de la taxonomía: las dificultades pueden deberse a malentendidos de los profesores sobre el significado de los niveles. Las subcategorías son:
 - a) Conocimiento superficial: incluye las creencias y las ideas preconcebidas que tienen los profesores sobre el significado de los niveles.
 - b) Distinta comprensión según su distinta experiencia: la interpretación de cada nivel de la taxonomía, así como el esfuerzo cognitivo son diferentes en un educador inexperto que en uno con experiencia. Las dificultades surgen cuando en un mismo grupo de evaluación existen educadores con distintas experiencias.
 - c) Su uso requiere un notable esfuerzo de memoria de los niveles.
- Complejidad del campo de la Informática: La dificultad del nivel cognitivo no solo viene dada por el contenido en estudio sino también por la herramienta que se use. Si bien las herramientas pueden tener el mismo fin, su complejidad puede variar, por lo que pueden clasificarse en distintos niveles de la taxonomía.

Soluciones encontradas para las dificultades

Por último, se presentan los resultados de las soluciones que identificamos. Algunos de los autores al encontrar una dificultad pueden o no haber aplicado algún tipo de medida para resolverla. Registramos cuatro artículos sin soluciones, tres artículos con dos soluciones y un artículo con tres. Por lo que el número de soluciones encontradas es igual a 16. Se estableció seis categorías de soluciones (véase la Tabla 10). La solución más frecuente es Dar pautas de aplicación (37,50%). La descripción de las categorías es la siguiente:

- Dar pautas de aplicación: los autores optaron por dar dos tipos de guías, la primera consiste en tomar programas e indicar a que nivel corresponden. Y la segunda plantea interpretar los términos de la taxonomía e indicar cómo emplearlos en Informática.
- Formación: capacitar en el uso de la taxonomía.
- Ampliar la taxonomía: son tres las soluciones que forman esta categoría, la primera propone aumentar la taxonomía con otras dimensiones como la complejidad y la dificultad, en la segunda se puede tener otro nivel de aplicación superior y la tercera sugiere utilizar conjuntamente con otras taxonomías p.ej. SOLO [10].
- Cambiar la terminología: recomiendan indicar términos relacionados con Informática para cada uno de los niveles de la taxonomía.
- Conocer el contexto educativo: saber el contexto de la pregunta dentro de enseñanza de una asignatura.
- Determinar el nivel cognitivo que los alumnos usarán: conocer el nivel cognitivo usado por la mayoría de los alumnos, en el contenido de la asignatura.

Tabla 8. Dificultades encontradas al usar la taxonomía de Bloom.

Categoría	Núm.	%	Cita(s) representativa(s)
Aplicabilidad para clasificar los objetivos o las tareas de evaluación	14	77,77%	<p>“While the complete analysis of all data across all participating institutions has not yet been completed, some initial insights and patterns have been observed. Firstly, categorising programming MCQ’s by cognitive complexity applying Bloom’s taxonomy, has proven challenging even to an experienced group of programming educators. This may suggest some deficiencies in the Bloom taxonomy when applying it to programming problems, or be a manifestation of the authors current level of understanding of how to apply the taxonomy. It also indicates that assessing programming fairly and consistently is a complex and challenging task, for which programming educators lack clear frameworks and tools.” [11]</p> <p>“We realized that there are many sub-categories for each category presented by the taxonomy, as learning and background information are wide and varied. As such, we found that categorising a novice response was, possibly unsurprisingly, a difficult task which posed some interesting challenges. We also went on to find that it was difficult to categorise the question.” [14]</p>
Especificar de forma precisa el conocimiento objeto de cada objetivo de aprendizaje o prueba de evaluación	2	11,11%	<p>"Consider the subtle, yet significant distinction between the concept of “iteration” and the concept of “for loop”. “Iteration” (a more abstract concept) may be implemented in many ways including a “for loop” (a more concrete concept). Assume that the CS1 instructor has chosen to apply BT on the concept of “iteration”. Upon reaching the Synthesis level, the instructor inadvertently switches to “for loop” and carries on. “Iteration” at the Synthesis level would require students to be able to design a new way to perform a loop (most probably a graduate level topic). On the other hand, the concept of “for loop” at the Synthesis level requires students to be able to write for loops that perform a specific task (something clearly within the CS1 domain). Almost subconsciously, the instructor has switched to a lower-level concept that is more appropriate for CS1 at a higher Bloom level." [12]</p>
Aplicabilidad para medir el progreso del alumno	1	5,56%	<p>“We wanted to work with a strictly hierarchal taxonomy, enabling us to monitor students' progress, but one that matched the context of the study and its objectives. Intersecting the cognitive dimension with the knowledge dimension enriched the taxonomy, but we still felt that this enrichment did not cope with the difficulty just described” [13]</p>
Comprensibilidad	1	5,56%	<p>“One of the difficulties with using the cognitive hierarchy in a programming context is clarifying what it means to apply a process and/or to create a process.”[5]</p>
TOTAL	18	100%	

Tabla 9. Causas para tener dificultades al usar la taxonomía de Bloom.

Categoría	Total	%	Subcategoría	Núm.	Cita(s) representativa(s)
Necesidad de conocer el contexto educativo	8	30,77%	los alumnos utilicen distinto esfuerzo cognitivo para resolver un mismo ejercicio	4	<p>“The classification of questions according to Bloom’s taxonomy involved the most initial disagreement. Although all disagreements were resolved through discussion, the inter-rater reliability for the Blooms categories across both exams was 75%. One reason for this lower level of inter-rater reliability was a policy that a question could be classified as multiple levels in the Bloom’s hierarchy. The classification took into account that there may be distinct ways of reasoning about a problem. When distinct ways of reasoning about a problem could be identified and these ways of reasoning were at different levels in the Bloom’s taxonomy, the question was classified with multiple levels in the Bloom’s hierarchy. Although a particular Bloom’s category is assumed to include all lower categories, this coding with multiple Bloom’s categories helped document when there were multiple paths to the answer.” [15]</p> <p>“It should also be recognised that the actual cognitive process that is applied to a specific task will depend on the individual solving that task. A given task might require nothing more than recall (the lowest level of cognitive process) for one individual, but may require another individual to generate a new solution to a situation that they find novel (using the highest level of cognitive process).” [4]</p>
			conocer la forma de enseñar el contenido	4	<p>“The example highlights the tight dependence on knowledge of the teaching context to correctly classify exam questions using Bloom. That is, different assumptions about the teaching context, and different assumptions about how the student can derive the answer, lead to vastly different classifications (Knowledge and Synthesis are at opposite ends).” [4]</p>
Limitaciones de la taxonomía	7	26,92%	conjunto incompleto o inadecuado de niveles para las tareas de programación	5	<p>“While the Bloom taxonomy was appealing as a potential design and analysis tool, we felt that in our context it would be too general and not informative enough. For example, Creating is considered to be much more complex than Understanding, but can we really say that creating a simple project – whose goal is to move one sprite from one point to another – is cognitively complex than fully understanding the concept of concurrency?” [13]</p>
			conjunto solapado	1	<p>“While the Bloom taxonomy was appealing as a potential design and analysis tool, we felt that in our context it would be too general and not informative enough. For example, Creating is considered to be much more complex than Understanding, but can we really say that creating a simple project – whose goal is to move one sprite from one point to another – is cognitively complex than fully understanding the concept of concurrency?” [13]</p>

			concebida para evaluar y no para especificar objetivos	1	“One possible explanation for this difficulty is that BT was never meant for specification, only for assessment. However, as mentioned in [8], this difficulty exists even within assessment. Another possibility is task interference. BT requires considerable effort to remember, especially if one is not well versed in it. This memory load affects one’s ability to apply it well onto different topics (i.e., “at what level of the taxonomy should this topic be covered?”). Finally, it has been argued that BT is not a perfect model [7, 8]. In our opinion, although possibly imperfect, BT facilitates a greater level of granularity in course specifications, thus considerably improving course delivery and assessment.” [12]
Terminología	5	19,23%	terminología extraña para programación	2	“as the original Bloom’s Taxonomy was written to suit the education field generally. Programming subjects are quite different in that the questions we create are not based on the keywords used in Bloom’s Taxonomy, but are mostly based on snippets of code.” [14] “Firstly the Bloom’s taxonomy categories were not developed to relate to programming questions, and which “keyword” in each category to apply to the question at hand was not obvious.” [14]
			falta de ejemplos	3	“The examples given by the taxonomy’s authors are not easy to translate into the programming domain. In many cases the categories within the knowledge domain, did not readily fit with concepts and tasks required in computer programming.” [11]
Comprensión deficiente de la taxonomía	5	19,23%	conocimiento superficial	3	“These readings reveal that the use of Bloom’s Taxonomy in a computer science context has generally been inconsistent. A possible reason for this is that the computer science educators performing the Bloom’s classifications did not have a strong understanding and/or a common understanding of the theory.” [16] “Participant self-reflection and selfexplanation revealed other reasons as to why different people make different classifications, namely due to pre-conceived misunderstandings of the categories, or different interpretations about the complexity of tasks and sophistication required to solve them.” [4]
			distinta comprensión según su distinta experiencia, esfuerzo cognitivo necesario para aplicar la propia taxonomía (compleja)	1	“One reason for this could be the difficulty of determining the taxonomic level of the assessment without having an intimate knowledge of the way in which the material being assessed was taught. (This difficulty was identified by Bloom et al themselves). This could lead to a task that was taught explicitly to students, and thus should be regarded as testing application, being assessed as involving a higher level skill such as synthesis—or vice versa.” [6]
			su uso requiere un notable esfuerzo de memoria de los niveles	1	“One possible explanation for this difficulty is that BT was never meant for specification, only for assessment. However, as mentioned in [8], this difficulty exists even within assessment. Another possibility is task interference. BT requires considerable effort to remember, especially if one is not well versed in it. This memory load affects one’s ability to apply it well onto different topics (i.e., “at what level of the taxonomy should this topic be covered?”). Finally, it has been argued that BT is not a perfect model [7, 8].

					In our opinion, although possibly imperfect, BT facilitates a greater level of granularity in course specifications, thus considerably improving course delivery and assessment.” [12]
Complejidad del campo de la Informática	1	3,85%	conocimientos conceptuales y específicos de la informática (o algún subcampo), p.ej. con conceptos de alto y bajo nivel	1	“The first problem is the inherent complexity of the field or disciplinary context. For example, in the systems modeling field, comparing the number and nature of symbols in Data Flow Diagrams (DFDs) to those in the formal specification language Z makes evident the considerable disparity between them. Whereas DFDs have only a few graphical symbols, Z has a complex mathematical notation. These differences superimpose a degree of difficulty at all levels of the taxonomy, leading to a paradigm shift between these two examples.” [17]
TOTAL	26	100%		26	

Tabla 10. Soluciones propuestas frente a las dificultades encontradas al usar la taxonomía de Bloom.

Categoría	Núm.	%	Cita(s) representativa(s)
Dar pautas de aplicación	6	37,50%	"Our interpretation of these three Bloom categories in the context of CS education is as follows (using terms taken from the corresponding sub-categories of the cognitive dimension of the revised Bloom taxonomy [1]): Understanding is interpreted as the ability to summarize, explain, exemplify and classify CS concepts (among which are programming constructs), and to compare them. Applying is interpreted as the ability to execute algorithms or code: to track them and recognize their goals. Creating is interpreted as the ability to plan and produce programs or algorithms." [13]
Formación	3	18,75%	"We concluded that academics seeking to use Bloom's Taxonomy need to be trained. To this end, our contribution described in this paper is a computer science contextualized web-based tutorial on the Bloom Taxonomy with interactive examples, user self-explanation and self-reflection." [4] "We aim to improve on this by providing CS educators with a contextualised interactive tutorial to learn about the theory and to practise applying it in a CS context, which should lead to more consistent classifications among different classifiers." [16]
Ampliar la taxonomía	3	18,75%	"Hence we incorporated further dimensions to our taxonomy to explain the Instructor Level of Complexity posed by the question, and the Novice Level of Difficulty relating to how they answered the question. We found the combination of these three dimensions allowed us to more clearly classify the questions." [14] "We chose to create a taxonomy which is a combination of the two. First, in the context of our study and corresponding to the learning goals, only a subset of categories seemed to be relevant. From the SOLO taxonomy we chose to focus on the three intermediate categories of Unistructural, Multistructural and Relational. From the Bloom taxonomy we chose to focus on Understanding, Applying and Creating." [13]
Cambiar la terminología	2	12,50%	"Carefully choose the words that reflect the level of cognitive behaviour and include these in the objectives. Writing objectives requires practice. I found that the objectives could be greatly improved by reviewing the literature about writing objectives and referring to good examples" [18] "We have attempted to control this aspect in the examples in Table 1 by limiting the context in the illustrations of levels 3–4 through the use of the qualifiers ‘‘well-defined’’ and ‘‘clearly delimited’’ [17]
Conocer el contexto educativo	1	6,25%	"Once the teaching context was elucidated, we were able to agree on an appropriate cognitive category for the assessment task in question." [5]
Determinar el nivel cognitivo que los alumnos usarán	1	6,25%	"The context is critical for assessing the level of process that we think most students will require in order to answer a given question." [5]
TOTAL	16	100%	

La Tabla 11 muestra el número de dificultades, causas y soluciones encontradas en cada artículo.

Tabla 11. Número de dificultades, causas y soluciones encontradas por artículo.

Código artículo	Dificultades	Causas	Soluciones
1	1	0	2
2	1	2	2
3	2	3	3
6	1	5	0
9	2	2	2
10	1	1	0
12	1	0	1
15	1	1	1
22	1	2	1
27	1	3	1
30	1	3	0
32	1	1	0
35	2	1	1
37	1	1	1
39	1	1	1
Total	18	26	16

4 Resumen de Hallazgos

Podemos resumir los resultados de las preguntas de investigación:

- El 75% de los artículos seleccionados utilizaron la versión original de la taxonomía de Bloom y un 25% la versión revisada.
- Un 5% de los artículos seleccionados usan una nueva taxonomía que resulta de la combinación de las taxonomías de Bloom y SOLO, un 10% usa la taxonomía de SOLO además de Bloom y un 85% utilizó únicamente Bloom.
- La taxonomía fue usada mayoritariamente en la enseñanza de la programación en primer año de la carrera de Informática con 53'5% de las respuestas.
- Al sumar los porcentajes obtenidos en las diversas materias de programación (programación de primer año, estructuras de datos y algoritmos, y otros cursos de programación), el porcentaje sube hasta un 69'8% de las respuestas, lo que nos indica que la taxonomía de Bloom es ampliamente usada en la enseñanza de la programación.
- Son numerosos los artículos que mencionan que han usado la taxonomía de Bloom, pero no aclaran para qué ni cómo (36'6%). Entre los que explican el

uso que han hecho de ella, destaca evaluar la actividad de los estudiantes (46'3%).

- De los 15 artículos que indican el uso de la taxonomía, pero no explican como la usaron, el 86,7% (13 artículos) de ellos también no comentaron tener dificultades.
- De los 15 artículos que expresan haber tenido dificultades en el uso de la taxonomía de Bloom, el 93,3% de ellos, señalaron como principal dificultad el clasificar un elemento dentro de los niveles cognitivos, mientras que un 86.67% expuso las causas de las dificultades y un 73.33% explicó las soluciones adoptadas.
- Identificamos cuatro causas como las más relevantes para tener dificultades en el uso de la taxonomía de Bloom, estas son: la necesidad de conocer el contexto de enseñanza (30,77%), las deficiencias de la propia taxonomía (26,92%), la terminología de la taxonomía (19,23%) y la deficiente comprensión de la misma (19,23%).

5 Discusión

Los resultados de nuestra revisión sistemática coinciden con el grupo de trabajo de Fuller et al. [7] en que la taxonomía de Bloom es la más usada en informática y que se usa principalmente para evaluar la actividad de los alumnos. La materia donde se ha usado principalmente es en programación (69,8% de materias contabilizadas).

Menos de la mitad de los artículos analizados indican haber tenido dificultades, sin embargo, en nuestra experiencia existen dificultades en el uso de taxonomía de Bloom. La inexistencia de comentarios en los artículos, no significa que no hayan tenido dificultades los autores en el uso de la taxonomía.

La principal dificultad que encontraron los autores es la clasificación de un objetivo, un contenido o una prueba dentro de los niveles cognitivos.

Se observó que el número de causas identificadas superó al número de dificultades y soluciones encontradas. La razón de tener más número de causas es porque los autores dan varias causas posibles a un mismo problema. El caso que mejor ejemplifica esto es el trabajo de Whalley et al. [11], donde se registra cinco causas para un solo problema. También se identificó varias causas potenciales en donde vemos cuatro categorías principales: necesidad de conocer el contexto de enseñanza, deficiencias de la propia taxonomía, la terminología de la taxonomía y la deficiente comprensión de la misma.

Identificamos una variedad de soluciones, pero la solución más frecuente es “dar pautas de aplicación”, como lo realiza Thompson et al. [5] en su trabajo donde proponen dar ejemplos relacionados con la programación.

6 Conclusiones

Hemos presentado de forma detallada una revisión sistemática del uso de la taxonomía de Bloom en la enseñanza de la informática. Se ha incluido la metodología usada, los resultados especificados y comentados, así como una discusión de los mismos.

Agradecimientos. Este trabajo se ha financiado con los proyectos TIN2015-66731-C2-1 del Ministerio de Economía y Competitividad de España y S2013/ICE-2715 de la Comunidad Autónoma de Madrid.

Referencias

1. Bloom, B.S.; Engelhart, M.D.; Furst, E.J.; Hill, W.H.; Krathwohl, D.R.: *Taxonomy of Educational Objectives: Handbook I: Cognitive Domain*. Longmans Group Ltd. (1956)
2. Anderson, L.W.; Krathwohl, D.R.; Airasian, P.W.; Cruikshank, K.A.; Mayer, R.E.; Pintrich, P.R.; Raths, R.; Wittrock, M.C.: *A Taxonomy for Learning, Teaching and Assessing. A Revision of Bloom's Taxonomy of Educational Objectives*. Addison-Wesley Longman (2001)
3. ACM & IEEE Computer Society, The Joint Task Force on Computing Curricula (2013): *Computer Science curricula 2013 Computer Science Curricula 2013 – Curriculum Guidelines for Undergraduate Degree Programs in Computer Science*. <https://www.acm.org/education/CS2013-final-report.pdf>. Accedido el 24 de abril de 2017
4. Gluga, R.; Kay, J.; Lister, R.; Kleitman S.; Lever, T.: Over-confidence and confusion in using bloom for programming fundamentals assessment. *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education*, pp 147-152 (2012)
5. Thompson, E.; Luxton-Reilly, A.; Whalley, J.L.; Hu, M.; Robbins, P.: Bloom's taxonomy for CS assessment. *Proceedings of the tenth conference on Australasian computing education*, Vol. 78, pp. 155-161 (2008)
6. Johnson, C.G.; Fuller, U.: Is Bloom's taxonomy appropriate for computer science?. *Proceedings of the 6th Baltic Sea conference on Computing education research: Koli Calling 2006*, pp. 120-123 (2006)
7. Fuller, U.; Johnson, C.G.; Ahoniemi, T.; Cukierman, D.; Hernán-Losada, I.; Jackova, J.; Lahtinen, E.; Lewis, T.L.; Thompson, D.M.; Riedesel, C.; Thompson, E.: Developing a computer science-specific learning taxonomy. *Proceeding ITiCSE-WGR '07 Working group reports on ITiCSE on Innovation and technology in computer science education*, pp. 152-170 (2007)
8. Britto, R.; Usman, M.: Bloom's taxonomy in software engineering education: A systematic mapping study. *Proceedings of Frontiers in Education Conference on FIE (IEEE 2015)*, pp. 1-8 (2015)
9. Glaser, B.G.; Strauss, A.L.: *The discovery of grounded theory: Strategies for qualitative research*. Transaction publishers (2009)
10. Biggs, J.B.; Collis, K.F.: *Evaluating the Quality of Learning: The SOLO Taxonomy (Structure of the Observed Learning Outcome)*. Academic Press (1982)
11. Whalley, J.L.; Lister, R.; Thompson, E.; Clear, T.; Robbins, P.; Kumar, P.K.; Prasad, C.: An Australasian study of reading and comprehension skills in novice programmers, using the bloom and SOLO taxonomies. *Proceedings of the 8th Australasian Conference on Computing Education*, Vol. 52, pp. 243-252 (2006)

12. Starr, C.W.; Manaris, B.; Stalvey, R.H.: Bloom's taxonomy revisited: specifying assessable learning objectives in computer science. *ACM SIGCSE Bulletin*, Vol. 40, No. 1, pp. 261-265 (2008)
13. Meerbaum-Salant, O.; Armoni, M.; Ben-Ari, M.: Learning computer science concepts with scratch. *Proceedings of the Sixth international workshop on Computing education research (ICER'10)*, pp 69-76 (2010)
14. Shuhidan, S.; Halmilton, M.; D'souza, D.: A taxonomic study of novice programming summative assessment. *Proceedings of the Eleventh Australasian Conference on Computing Education*, Vol. 95, pp. 147-156 (2009)
15. Lewis, C.M.; Khayrallah, H.; Tsai, A.: Mining data from the AP CS a exam: patterns, non-patterns, and replication failure. *Proceedings of the ninth annual international ACM conference on International computing education research*, p. 115-122 (2013)
16. Gluga, R.; Kay, J.; Lister, R.; Kleitman, S.: Mastering cognitive development theory in computer science education. *Computer Science Education*, Vol. 23, No. 1, pp. 24-57 (2013)
17. Oliver, D.; Dobebe, T.; Greber, M.; Roberts, T.: Comparing course assessments: When lower is higher and higher, lower. *Computer Science Education*, Vol. 14, No. 4, pp. 321-341 (2010)
18. Box, I.: Object-oriented analysis, criterion referencing, and Bloom. *Proceedings of the Sixth Australasian Conference on Computing Education*, Vol. 30, pp. 1-8 (2004)

Apéndice A: Lista de los artículos seleccionados para la revisión sistemática.

En la siguiente lista se presentan los artículos seleccionados. La numeración corresponde al código de identificación para cada documento. No representa ningún orden en particular.

1. Box, I.: Object-oriented analysis, criterion referencing, and Bloom. *Proceedings of the Sixth Australasian Conference on Computing Education*, Vol. 30, pp. 1-8 (2004)
2. Shuhidan, S.; Halmilton, M.; D'souza, D.: A taxonomic study of novice programming summative assessment. *Proceedings of the Eleventh Australasian Conference on Computing Education*, Vol. 95, pp. 147-156 (2009)
3. Thompson, E.; Luxton-Reilly, A.; Whalley, J.L.; Hu, M.; Robbins, P.: Bloom's taxonomy for CS assessment. *Proceedings of the tenth conference on Australasian computing education*, Vol. 78, pp. 155-161 (2008)
4. Lister, R.: On blooming first year programming, and its blooming assessment. *Proceedings of the Australasian conference on Computing education ACM*, pp. 158-162 (2000)
5. Lister, R.; Leaney, J.: First year programming: let all the flowers bloom. *Proceedings of the fifth Australasian conference on Computing education*, Vol. 20, Australian Computer Society Inc., pp. 221-230 (2003)
6. Whalley, J.L.; Lister, R.; Thompson, E.; Clear, T.; Robbins, P.; Kumar, P.K.; Prasad, C.: An Australasian study of reading and comprehension skills in novice programmers, using the bloom and SOLO taxonomies. *Proceedings of the 8th Australasian Conference on Computing Education*, Vol. 52, pp. 243-252 (2006)
7. Schulte, C.; Bennedsen, J.: What do teachers teach in introductory programming? *Proceedings of the second international workshop on Computing education, ACM*, pp. 17-28 (2006)
8. Fitzgerald, S.; Simon, B.; Thomas, L.: Strategies that students use to trace code: an analysis based in grounded theory. *Proceedings of the first international workshop on Computing education, ACM*, pp. 69-80 (2005)
9. Meerbaum-Salant, O.; Armoni, M.; Ben-Ari, M.: Learning computer science concepts with scratch. *Proceedings of the Sixth international workshop on Computing education research (ICER'10)*, pp 69-76 (2010)
10. Lewis, C.M.; Khayrallah, H.; Tsai, A.: Mining data from the AP CS a exam: patterns, non-patterns, and replication failure. *Proceedings of the ninth annual international ACM conference on International computing education research*, p. 115-122 (2013)
11. Kaloti-Hallak, F.: The effect of robotics activities on students' learning and attitudes, *Proceedings of the tenth annual conference on International computing education, ACM*, pp. 153-154 (2014)
12. Taylor, B.; Kaza, S.: Security injections: modules to help students remember, understand, and apply secure coding techniques, *Proceedings of the 16th annual joint conference on Innovation and technology in computer science education*, pp. 3-7 (2011)

13. Honing, W.L.: Teaching and assessing programming fundamentals for non majors with visual programming, *Proceedings of the 18th ACM conference on Innovation and technology in computer science education*, pp. 40-45 (2013)
14. Walker, G. N.: Experimentation in the computer programming lab, *Proceeding ITiCSE-WGR '04 Working group reports from ITiCSE on Innovation and technology in computer science education*, pp. 69-72 (2004)
15. Naps, T.L.; Rössling, G.; Almstrum, V.; Dann, W.; Fleischer, R.; Hundhausen, C.; Korhonen, A.; Malmi, L.; McNally, M.; Rodger, S.: Exploring the role of visualization and engagement in computer science education, *Proceeding ITiCSE-WGR '02 Working group reports from ITiCSE on Innovation and technology in computer science education*, pp. 131-152 (2002)
16. Fuller, U.; Johnson, C.G.; Ahoniemi, T.; Cukierman, D.; Hernán-Losada, I.; Jackova, J.; Lahtinen, E.; Lewis, T.L.; Thompson, D.M.; Riedesel, C.; Thompson, E.: Developing a computer science-specific learning taxonomy. *Proceeding ITiCSE-WGR '07 Working group reports on ITiCSE on Innovation and technology in computer science education*, pp. 152-170 (2007)
17. Serrano Camara, L. M.; Paredes Velasco, M.; Velazquez-Iturbide, J. Á.: Evaluation of a collaborative instructional framework for programming learning. *Proceedings of the 17th ACM annual conference on Innovation and technology in computer science education*, pp. 162-167 (2012)
18. Urquiza-Fuentes, J.; Velazquez-Iturbide, J. A.: Comparing the effectiveness of different educational uses of program animations. *Proceedings of the 17th ACM annual conference on Innovation and technology in computer science education*, pp. 174-179 (2012)
19. Mohamed Shuhidan, S.; Hamilton, M.; D'souza, D.: Understanding novice programmer difficulties via guided learning. *Proceedings of the 16th annual joint conference on Innovation and technology in computer science education*. pp. 213-217 (2011)
20. Magnenat, S.; Shin, J.; Riedo, F.; Siegwart, R.; Ben-Ari, M.: Teaching a core CS concept through robotics, *Proceedings of the 2014 conference on Innovation & technology in computer science education 2014, ACM*, pp. 315-320 (2014)
21. Cukierman, D.; Thompson, D.M.: Learning strategies sessions within the classroom in computing science university courses. *Proceeding ITiCSE '07 Proceedings of the 12th annual SIGCSE conference on Innovation and technology in computer science education*, pp. 341-341 (2007)
22. Johnson, C.G.; Fuller, U.: Is Bloom's taxonomy appropriate for computer science?. *Proceedings of the 6th Baltic Sea conference on Computing education research: Koli Calling 2006*, pp. 120-123 (2006)
23. Buck, D.; Stucki, D.J.: JKarelRobot: A Case Study in Supporting Levels of Cognitive Development in the Computer Science Curriculum. *Proceeding SIGCSE '01 Proceedings of the thirty-second SIGCSE technical symposium on Computer Science Education*, pp. 16-20 (2001)
24. Miller, L.D.; Soh, L.; Neilsen, B.; Kupzyk, K.; Samal, A.; Lam, E.; Nugent, G.: Revising computer science learning objects from learner interaction data. *Proceedings of the 42nd ACM technical symposium on Computer science education*, pp. 45-50 (2011)
25. Köppe, C.; Pruijt, L.: Improving students' learning in software engineering education through multi-level assignments. *Proceedings of the Computer Science Education Research Conference*, pp. 57-62 (2014)
26. Buck, D.; Stucki, D.J.: Design early considered harmful: graduated exposure to complexity and structure based on levels of cognitive development. *Proceeding SIGCSE '00 Proceedings of the thirty-first SIGCSE technical symposium on Computer science education*, pp. 75-79 (2000)
27. Gluga, R.; Kay, J.; Lister, R.; Kleitman S.; Lever, T.: Over-confidence and confusion in using bloom for programming fundamentals assessment. *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education*, pp 147-152 (2012)

28. Doran, M.V.; Langan, D.D.: A cognitive-based approach to introductory computer science courses: lesson learned. *Proceeding SIGCSE '95 Proceedings of the twenty-sixth SIGCSE technical symposium on Computer science education*, pp. 218-222 (1995)
29. Howard, R.A.; Carver, C.A.; Lane, W.D.: Felder's learning styles, Bloom's taxonomy, and the Kolb learning cycle: tying it all together in the CS2 course. *Proceeding SIGCSE '96 Proceedings of the twenty-seventh SIGCSE technical symposium on Computer science education*, pp. 227-231(1996)
30. Starr, C.W.; Manaris, B.; Stalvey, R.H.: Bloom's taxonomy revisited: specifying assessable learning objectives in computer science. *ACM SIGCSE Bulletin*, Vol. 40, No. 1, pp. 261-265 (2008)
31. Soh, L.K.; Samal, A.; Person, S.; Nugent, G.; Lang, J.: Designing, implementing, and analyzing a placement test for introductory CS courses. *Proceeding SIGCSE '05 Proceedings of the 36th SIGCSE technical symposium on Computer science education*, pp. 505-509 (2005)
32. Petersen, A.; Craig, M.; Zingaro, D. Reviewing CS1 exam question content. *Proceedings of the 42nd ACM technical symposium on Computer science education*, pp. 631-636 (2011).
33. Malloy, J.; Burge, J.: SEURAT Edu: A Tool to Assist and Assess Student Decision-Making in Design. *Proceedings of the 47th ACM Technical Symposium on Computing Science Education*, pp. 669-674 (2016)
34. Dolog, P.; Thomsen, L.L.; Thomsen, B.: Assessing Problem-Based Learning in a Software Engineering Curriculum Using Bloom's Taxonomy and the IEEE Software Engineering Body of Knowledge. *ACM Transactions on Computing Education (TOCE)*, Vol. 16, No. 3, (2016)
35. Oliver, D.; Dobele, T.; Greber, M.; Roberts, T.: Comparing course assessments: When lower is higher and higher, lower. *Computer Science Education*, Vol. 14, No. 4, pp. 321-341 (2004)
36. Hamouda, S.; Shaffer, C.A.: Crib sheets and exam performance in a data structures course. *Computer Science Education*, Vol. 26, No.1, pp.1-26 (2016)
37. Alaoutinen, S.: Evaluating the effect of learning style and student background on self-assessment accuracy. *Computer Science Education*, Vol. 22, No. 2, pp. 175-198 (2012)
38. Port, D.; Boehm, B.: Introducing risk management techniques within project based software engineering courses. *Computer science education*, Vol. 12, No. 1-2, pp. 37-55 (2002)
39. Gluga, R.; Kay, J.; Lister, R.; Kleitman, S.: Mastering cognitive development theory in computer science education. *Computer Science Education*, Vol. 23, No. 1, pp. 24-57 (2013)
40. Manaris, B.; Wainer, M.; Kirkpatrick, A.E.; Stalvey, R.H.; Shannon, C.; Leventhal, L.; Barnes J.; Wright J.; Schafer J.B.; Sanders, D. Implementations of the CC' 01 human-computer interaction guidelines using Bloom's taxonomy. *Computer Science Education*, Vol. 17, No. 1, 21-57 (2007)