Bloom's Taxonomy to Improve Teaching-Learning in Introduction to Programming

S ónia Rolland Sobral

Abstract—**Students in computer science courses entering higher education begin with computer thinking and programming languages in a curricular unit (CU) that can be referred to in various ways, like CS1 or programming fundamentals. This CU is very important for the academic and professional path of those students. Teachers and those responsible for those courses must carefully define the learning objectives, the learning strategies and the assessment of this teaching-learning. Bloom's taxonomy, in its different variations, is a powerful tool that helps in these tasks and that gives clear indications on the language that is to be used - which is useful for perceiving both the level of colleagues and the level of accreditations and assessments of courses. This article provides a detailed description of Bloom's taxonomy and its changes over the years. Studies carried out in the context of teaching fundamentals of programming and using Bloom's taxonomy are listed and analyzed. In the end, the conclusions and the definition of future works are made.**

Index Terms—**Bloom's taxonomy, CSE, CS1, introduction to programming.**

I. Introduction

Introduction to programming for first year students is a very important course for the academic and professional path of those who enter a higher education institution and a computer science course. The content of these units, the objectives, the programming languages and the way everything is learned and taught need a lot of thought to be successful. In these CUs, students start in the world of computational thinking, use tools for specification and learn one or more programming languages. The teacher must identify what he wants to accomplish in the teaching process [1]. It is clear that it is easier to achieve goals when they are well defined, however it is more difficult for students to reach a higher level of cognitive development, as they do not know exactly what is expected of them during and after the teaching process. The clear and structured definition of instructional objectives, considering the acquisition of knowledge and skills appropriate to the professional profile to be formed, will direct the teaching process towards the appropriate choice of strategies, methods, delimitation of specific content, assessment instruments and, consequently, for effective and lasting learning [2]. And because teaching is an intentional (for some purpose) and reasoned act it is imperative that the objectives are well defined [3].

Bloom's Taxonomy [4] was originally presented in a document aimed at improving the communication between teachers. In this document, the three domains of learning were structured as following: cognitive, psychomotor and affective domain (knowledge, skills and attitudes respectively). Bloom's Taxonomy was later revised [3], [5] to form a table with the knowledge dimension and the cognitive process dimension. The table also appeared to be a simplifier for those who must clearly structure the educational objectives, helping the instructors to plan better, in addition to helping in the appropriate choice of strategies. It is important to note the accreditation criteria which, although different for each country, requires departments to provide an accurate description of educational programs, as well as the results of using these programs.

Bloom's taxonomy helps not only to create very clear learning objectives and course materials in terms of language and specification, but also to connect with the entire evaluation part of those objectives.

There are some investigations that explore the use of Bloom's taxonomy applied to teaching-learning in the introduction to programming UC facing the cycle of planning, teaching, learning and assessing. These articles have various types of approaches and describe different ways in which these taxonomies have been applied. They are generally directed towards course design and assessment. There are others which have an approach to the design of teaching and assessment materials. Other studies are focused on students and on the analysis of students' responses to exercises.

Each student learns differently and at their own pace. There are two perspectives; assuming that under the same learning conditions everyone would learn the content with the same competence and depth [6] or assuming that in the same teaching conditions learning differs in terms of depth and abstraction of the acquired knowledge [7]. One of the problems of teaching and learning is this: defining equal rhythms for everyone, which can be a success for some students and a bad experience for others. It is important to distinguish the terms "learning style", "cognitive style" and "learning strategy" which are often used imprecisely [8]. The term "learning style" refers to the mode of instruction or study that is most effective for each individual. It is suggested that instruction is provided in a format which matches the preferences of each student [9]. Cognitive styles are defined as habitual and stable strategies that determine the ways of perceiving, remembering, thinking and solving problems. The cognitive style reflexivity-impulsivity refers to preference shown by people to respond quickly versus people who prefer to do so slowly in order to reduce the number of errors in problem solving tasks [10]. In the context of school learning, the term learning strategy is currently used as a generic term to designate all the behaviors adopted by the

Manuscript received April 22, 2020; revised November 6, 2020.

S ónia Rolland Sobral is with the Universidade Portucalense, Portugal (e-mail: soniarollandsobral@gmail.com).

doi: 10.18178/ijiet.2021.11.3.1504

learner in the process of learning, and all that can influence the way in which he will learn how to do it [11]. When a teacher creates the learning objectives for a course, he must take into account who his students are and that they learn differently. Teaching-learning strategies must be clear and precise, being directed towards learning objectives and providing for different learning styles.

This article provides a detailed description of Bloom's taxonomy and its changes over the years. Studies carried out in the context of teaching fundamentals of programming and using Bloom's taxonomy are listed and analyzed. In the end, the conclusions and the definition of future works are made.

II. Bloom's Taxonomy

In 1948, at an informal meeting of university examiners who attended the American Psychological Association Convention in Boston, the idea to create a classification system that could be used to facilitate communication between examiners arose. At the beginning of the discussions, the question whether educational objectives may or may not be classified arose too. Between 1949 and 1953 a group of university professors and researchers participated in conferences whose contribution resulted in a document (Taxonomy of educational objectives: The Classification of Educational Goals, 1959) [4] whose editor was Professor Benjamin S. Bloom of University of Chicago. This document presents a taxonomy that became known as Bloom's Taxonomy, clarifying the three domains of learning as following: cognitive, psychomotor and affective domain (or knowledge, skills and attitudes). The Original Bloom's Taxonomy provided carefully developed definitions for each of the six major categories in the cognitive domain: Knowledge, Comprehension, Application, Analysis, Synthesis, and Evaluation. The categories were ordered from simple to complex and from concrete to abstract: lower-order skills that require less cognitive processing to higher-order skills that require deeper learning and a greater degree of cognitive processing [12].

In 2001, the review report was published in a book entitled "A taxonomy for learning, teaching, and assessing: a revision of Bloom's taxonomy of educational objectives" by Lorin W. Anderson, David R. Krathwohl, Benjamin Samuel Bloom. In 2002, David R. Krathwohl published a new document, A Revision of Bloom's Taxonomy: An Overview [5]. The revised Taxonomy is a two-dimensional framework: Knowledge and Cognitive Processes. The six categories of the original Taxonomy were renamed: Knowledge to Remember, Comprehension to Understand, Synthesis to Create; Application, Analysis, and Evaluation changed to their verb forms: Apply, Analyze, and Evaluate (Fig. 1).

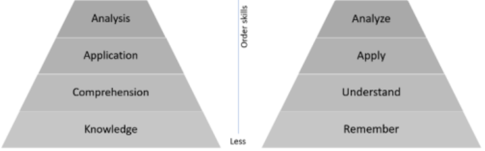

Fig. 1. Original and revised bloom's taxonomy, cognitive domain.

The Structure of the Cognitive Process Dimension of the Revised Taxonomy is defined as following:

- Remember: Retrieving relevant knowledge from long-term memory
  a) Recognizing (or Identifying)
  b) Recalling (or Retrieving)
- Understand: Determining the meaning of instructional messages, including oral, written, and graphic communication
  a) Interpreting (or Clarifying, paraphrasing, representing, translating)
  b) Exemplifying (or illustrating, instantiating)
  c) Classifying (or Categorizing, subsuming)
  d) Summarizing (or Abstracting, generalizing)
  e) Inferring (or Concluding, extrapolating, interpolating, predicting)
  f) Comparing (or Contrasting, mapping, matching)
  g) Explaining (or Constructing models)
- Apply: Carrying out or using a procedure in a situation
  a) Executing (or Carrying out)
  b) Implementing (or Using)
- Analyze: Breaking material into its constituent parts and detecting how the parts relate to one another and to an overall structure or purpose
  a) Differentiating (or Discriminating, distinguishing, focusing, selecting)
  b) Organizing (or Finding coherence, integrating, outlining, parsing, structuring)
  c) Attributing (or Deconstructing)
- Evaluate: Making judgments based on criteria and standards
  a) Checking (or Coordinating, detecting, monitoring, testing)
  b) Critiquing (or Judging)
- Create: Putting elements together to form a novel, coherent whole or make an original product
  a) Generating (or Hypothesizing)
  b) Planning (or Designing)
  c) Producing (or Constructing).

The Structure of the Knowledge Dimension of the Revised Taxonomy define:

- Factual Knowledge: The basic elements that students must know to be acquainted with a discipline or solve problems in it (Knowledge of terminology, Knowledge of specific details and elements)
- Conceptual Knowledge: The interrelationships among the basic elements within a larger structure that enable them to function together (Knowledge of classifications and categories, Knowledge of principles and generalizations, Knowledge of theories, models, and structures)
- Procedural Knowledge: How to do something; methods of inquiry, and criteria for using skills, algorithms, techniques, and methods (Knowledge of subject-specific skills and algorithms, Knowledge of subject-specific techniques and methods, Knowledge of criteria for determining when to use appropriate procedures)
- Metacognitive Knowledge: Knowledge of cognition in general as well as awareness and knowledge of one's own cognition (Strategic knowledge, Knowledge about cognitive tasks, including appropriate contextual and

conditional knowledge, Self-knowledge).

The Taxonomy Table has in vertical axis the Knowledge dimension and in horizontal axis the Cognitive Process dimension (Table I).

TABLE I: TAXONOMY TABLE

	Remember	Understand	Apply	Analyze	Evaluate	Create
Factual Knowledge						
Conceptual Knowledge						
Procedural Knowledge						
Metacognitive Knowledge						

Each interception in the table is equivalent to the intended learning objectives and includes some subject content and a description of what should be done with or for that content. The objectives are defined in a content and in a cognitive process: respectively a noun (or a substantive phrase) and a verb (or a verbal phrase) [5]. Krathwohl's example is that a student should be able to remember the law of supply and demand in economics. So we will have "The student shall be able to" (or "the student will learn"), the substantive sentence is "Law of supply and demand" and the verb "remember", that is "The student shall be able to remember the economics law of supply and demand".

To fill the cells in the table, the intercept of the horizontal axis with the vertical axis is used. A simple example of an objective that can be placed right in the first cell is remembering the name of the editor of the initial Bloom Taxonomy document. In the Structure of the Knowledge Dimension we will have factual knowledge (in this case Knowledge of specific details and elements). In the Structure of the Cognitive Process Dimension we will have Remember. So, the objective is placed in the cell from top to left, that is, in the interception of Remember with factual knowledge. The table is then filled with the objectives. Obviously, cells (or interceptions) are only filled when they are applied, leaving in blank those cells that do not make sense in concrete learning. The table is intended to be a facilitator to clearly structure educational objectives, helping instructors to plan better, as well as helping in the appropriate choice of strategies.

III. APPLICATIONS OF BLOOM'S TAXONOMY TO COMPUTER SCIENCE

Bloom's taxonomy is broadly defined. Each discipline must use it and apply it in order to contextualize this knowledge. There are several examples of Bloom's taxonomy applied to disciplines, for example to biology [13], management [14] or music [15]. Some authors write about the application of Bloom's theory to computer science. In the study by Susana Masapanta-Carrión and J. Ángel Velázquez-Iturbide [16] for the use of Bloom's Taxonomy in Computer Science Education, they found that 46% of the publications used Bloom's taxonomy while regarding student evaluation and 37% of the publications mention that they used or were based on Bloom's taxonomy, but without explaining how. The authors defined the following sub-categories: evaluate students, where taxonomy is used to measure students' knowledge (development of questions or problems, rating previously developed questions or problems at cognitive levels, rating student performance on cognitive levels), program instructions, specify the learning objectives for a course, create a new taxonomy, or develop educational software.

There are several types of courses within computer science education that use Bloom's taxonomy. Michael V. Doran and David D. Langan [17] report a cognitive based approach to implement Computing I and II courses, explicitly defining and scheduling course micro-objectives, mapping each micro-objective to a specific Bloom knowledge level, and helping to achieve and measure those objectives. Philip Machanick [18] presents the application of Bloom's taxonomy for the design of three different computer science courses: Data and Data Structures, Algorithms and Artificial Intelligence and Computer Architecture.

Johnson and Fuller [19] checked whether the Bloom taxonomy is appropriate for computing. They conducted a study that analyzed the 54 evaluations that were given to students by a panel of five academics who were asked to decide which of Bloom's taxonomy levels were being assessed for that specific assessment. The study found a considerable disagreement between the academics responsible for the modules and the evaluators on the level at which the evaluation was being carried out: the evaluators felt that the vast majority of the assessment was at the application level, while the academics responsible considered that they were also evaluating analysis (there is difficulty in determining the taxonomic level of the assessment without having an intimate knowledge of the way in which the material being evaluated has been taught). It was also found that it is not clear if the two highest levels of Bloom's taxonomy, synthesis and evaluation, should or not be applied: on one hand, there is the opinion that they should not be covered until the last year of the degree program, while for others these levels are suitable because they are of core importance, so they must be taught and evaluated. Similarly, Sanders and Mueller [20] argue that CUs in the early years of the program should focus on reaching the lowest levels of Bloom's taxonomy, leaving the highest levels for final years.

Essi Lahtinen [21] presents a study of 254 undergraduate students of a basic programming course whose performance was measured on different levels of Bloom's Taxonomy and the results of the statistical analysis of clusters that suggest that the groups of students obtained align with Bloom's taxonomy were compiled: underperforming students at lower levels may still perform well at higher levels of taxonomy. Therefore, Lahtinen suggests six groups of students that the teacher should recognize when organizing basic programming courses: Competent students, Practical students, Unprepared students, Theoretical students, Memorizing students and Indifferent students. As an example, the theoretical students performed well on the highest level (evaluation) while having problems with application and synthesis.

Starr and Colleagues [22] used a meta-level structure for Bloom's taxonomy, with differences between producing and explaining at three levels: Beginner, Intermediate and Expert. Recall and Comprehension at the lowest level, Application and Analysis at the intermediate level and Synthesis and Evaluation at the highest level (respectively to produce and explain).

A Bloom rating [23] was created for six courses by Oliver and Colleagues, including Programming 1. This course appears with an evaluation level of 0% but has a high score for synthesis. Interestingly, the knowledge and application levels go from programming courses 1 to programming 3.

Shneider and Gladkikh [24] suggest a generic approach to planning questioning strategies based on application of Bloom's cognitive taxonomy to a variety of different subjects, including Programming with Visual Basic, and define problem models useful at different levels of student assessment. They used a modified taxonomy (MT), changing Remembering to Recall (Re), Understanding to Comprehension (Co), Applying to Application (Ap), Analysing to Analysis (An), Evaluating to Evaluation (Ev) and Creating to Synthesis (Sy). They found it to be a good practice to arrange the learning outcomes in an order corresponding to the levels of the model. Some outcomes would relate to more than one level. As an example, they give a set of learning results using the topic Arrays of the Programming with Visual Basic course that we put in table format (Table II).

TABLE II: LEARNING RESULTS USING THE TOPIC ARRAYS OF THE PROGRAMMING WITH VISUAL BASIC COURSE [24]

	Re	Co	Ap	An	Ev	Sy
Define the terms and concepts related to the array structures	X					
Establish an array and refer to individual elements in the array		X	X			
Use repetition structures to initialize an array and traverse its elements			X			X
Perform sorting of array elements				X	X	X
Perform calculations and data manipulations using arrays			X	X		
Merge two ordered arrays			X			X
Bind arrays to GUI objects					X	X
Store and search data in multidimensional arrays				X	X	X

Several examples of the questioning model for formative assessment were given by the authors. From these examples we remove one from each level of the modified taxonomy (MT) and put it in the following table (Table III).

TABLE III: SOME EXAMPLES OF THE QUESTIONING TEMPLATE FOR THE FORMATIVE ASSESSMENT [24]

MT	Problem type	Provided input	Expected form of solution
Re	Define the given terms	Concept or term	Description
Co	Illustrate passing an array to a function		An example with a short explanation
Ap	Write a procedure to place the numbers from a given array into the other arrays	Array definition	Code snippet
An	Find the error in the given block of code	A block of code containing an array	A clear indication of an error with an explanation
Ev	Discuss, giving reasons, which type of search would be best for the given array	Definition of array	Discussion with reasons
Sy	Reconstruct the Select Case block in a fragment of code so that all elements of two arrays are merged into the third array	A program segment	Code snippet

Lister and Leaney [25] use a two-pronged solution: first, different tasks are defined according to the capacity of the students and, in the second part, with differences in the nature of the tasks reflected in different levels of Bloom taxonomy. They define that weaker students are required to demonstrate knowledge and understanding (must be able to write modest but complete programs and to understand the issues of a good program design, but without always making the connection between these problems and the modest programs they write), average students' should exhibit application and analysis level performance at program design and high-level students must have performance at the “synthesis” and “evaluation” levels (conceiving / building their own project, and criticizing other students' projects, being aware of good software design principles).

Howerd and Collegues [26] found Merge and Quick Sorts Lesson Objectives in Terms of Bloom’s Taxonomy:

1) The student will know how the merge sort works.
2) The student will know how the quicksort works.
3) The student will comprehend the Big O analysis of merge sort and quicksort.
4) The student will evaluate the five sorting methods covered in Kruse, Chapter 7, for both contiguous and linked lists in terms of Storage Space, Running Time and Programming Effort.

Errol Thompson and colleagues [27], despite finding that Bloom's taxonomy is difficult to consistently apply to assessment tasks in introductory programming courses, consider it a valuable tool for analysis and discussion of program evaluation. They provide an interpretation of Bloom's Taxonomy’s review for computer science because they feel that it is important for the computer science discipline to look carefully at the cognitive processes that programming requires. They gave some examples, of which we list some:

Remember: List the arithmetic operators in increasing order of precedence.

Understand: Explaining an algorithm

Apply: Evaluate the expression: 2 + 4 / 7 * 5 % 3 == 7.

Analyze: Given the code for a Circle class, the students are asked: What is the method Circle in this class?

Evaluate: Discuss the differences between these solutions

```
private double numbers[] = new double[10];
private int used = 0;
for (int i = 0; i < used; i++) {
 min = Math.min(min, numbers[i]);
```

And

```
public double min() {
 double min = numbers[0];
 for (double number : numbers) {
 min = Math.min(min, number);
 }
 return min;
}
```

Create: Write a method get24HourTime() which accepts three parameters and returns a String. The three parameters are an int representing the hour value, anint representing the minute value and a String which is either “am” or “pm”.

Scott [28] gives examples from computer science of each Bloom category:

- Name the three kinds of loops found in C++. (Remember)
- Give the output of this code that contains a switch statement. (Understand)
- Convert a for loop to a while loop. (Apply)
- Explain the steps you would perform to find the average of values in an array. Do not write code, explain in English. (Analyze)
- The following code is supposed to find the Lowest Common Denominator. Give test data that will test this code as completely as possible. For each piece of test data, explain why it is included and what it is testing. (Evaluate)
- Write code to find the sum of all primes up to a value that is typed in at the keyboard. Break the program up into subprograms. (Create).

Using the modified taxonomy, Christopher W. Starr, Bill Manaris, and RoxAnn H. Stalvey [22] gave a brief explanation of Bloom’s levels, and sample questions for the concept “computer program”:

1) Recall: The student is expected to recite memorized information about the concept. "What is a program?"
2) Comprehension: The student is expected to explain the concept in his or her own words. "How is a program similar to a recipe?"
3) Application: The student is expected to apply the concept to a situation. "What is the output of this program?"
4) Analysis: The student is expected to separate materials or concepts into component parts so that their organizational structure may be understood. "Create a top down design for a program to perform a given task."
5) Synthesis The student is expected to put parts together to form a whole, with emphasis on creating a new meaning or structure. "Write a program to perform a given task."
6) Evaluation The student is expected to make judgments about the value of ideas or materials. "Given two programs that perform the same task, which one is better and why?"

Some authors use taxonomy with a student-centered perspective. Jacqueline L. Whalley and Colleagues [29] report a study of reading and comprehension skills in novice programmers, developing a set of ten questions within two key pedagogical frameworks: Bloom and SOLO taxonomies. Four questions were Cognitive Process Category Apply (sub-category Executing), four Understand (Comparing) and two Analyze (Differentiating). The study proved to be challenging even for an experienced group of programming educators, suggesting shortcomings in Bloom's taxonomy when applying it to programming problems. They also found as a result that evaluating the programming fairly and consistently is a complex and challenging task, for which programming educators do not have clear structures and tools.

Essi Lahtinen and Tuukka Ahoniemi [30] use each level of Bloom Taxonomy for Visualizations to Support Programming. Diana Cukierman and Donna McGee Thompson [31] report their investigation of the impact of integration instruction of learning strategies within regular laboratory time courses to help students learn. In the initial session, Learning Strategies, and students were introduced to the taxonomy of Bloom's learning levels and demonstrated how this model applies to course topics.

Ursula Fuller and Colleagues [32] recommended the use of a matrix that separates the six levels of Bloom into two dimensions, Production (incorporation of applying and creating) and interpretation (incorporating remember, understand, analyze and evaluate). The purpose of the matrix is to identify a variety of different learning trajectories and therefore, guide students on how to improve their skills and understanding.

IV. Conclusions and Future Work

There is a need for tools that help the teacher to clearly define instructional objectives, as well as assessment principles. The language used must be in a way that others understand it, and that it is in accordance with the principles of accreditation of courses (even if these are different in each country and have changed over the years). Bloom's taxonomy seems to be a good alternative to be used.

In this article we study the original and revised Bloom taxonomy. We analyze the studies done in the area of computer science, sub-area of introduction to programming, fundamentals of programming or, as it is widely called, CS1. We found that there are different approaches to taxonomy: while many researchers use the tool in order to clearly define learning objectives, others use it to measure students' knowledge, others to build study materials and assessment tests, while others use taxonomy so that students clearly understand what is intended with the course. There are some differences between the studies, namely regarding the sub-types of students: some define each level of students according to what is defined in Bloom's taxonomy, others find profiles of some that are not consonant with the six categories of the cognitive domain of taxonomy.

As future work we intend to use the topics and learning objectives defined in the introductory units of the ACM / IEEE Computer Science Curriculum recommendation [33] in order to be able to define the students' levels of learning and assessment, while trying to assess what they are and what categories are used in the cognitive domain of Bloom's taxonomy. Although some researchers propose a new and modified taxonomy, it seems to us that Bloom's taxonomy fits what is intended in relation to curricular units of introduction to programming that are the focus of our investigation.

Conflict of Interest

The authors declare no conflict of interest

Author Contributions

The author did a literature review, defined the methodology, research and data treatment, data analysis and conclusions, having written the entire document.

REFERENCES

[1] C. A. Vaughan, "Identifying course goals: Domains and levels of learning," *Teaching Sociology*, vol. 7, no. 3, pp. 265-279, 1980.

[2] A. Ferraz and R. BelhotI, "Taxonomia de Bloom: revisão teórica e apresentação das adequações do instrumento para definição de objetivos instrucionais," *Gestão & Produção*, vol. 17, no. 2, 2010.

[3] L. Anderson, D. Krathwohl, and B. Bloom, "A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives," 2001.

[4] B. Bloom, M. Engelhart, E. Furst, W. Hill, and D. Krathwohl, "Taxonomy of educational objectives: The classification of educational goals," David McKay, 1956.

[5] D. Krathwohl, "A revision of bloom's taxonomy: An overview," *Theory into Practice*, vol. 41, no. 4, pp. 212-218, 2002.

[6] B. Bloom, "Some major problems in educational measurement," *Journal of Educational Research*, vol. 38, no. 2, pp. 139-142, 1944.

[7] B. Bloom, J. Hastings, and G. Madaus, *Handbook on Formative and Summative Evaluation of Student Learning*, McGraw-Hill, 1971.

[8] S. Cassidy, "Learning styles: An overview of theories, models, and measures," *Educational Psychology*, vol. 24, no. 4, pp. 419-444, 2004.

[9] H. Pashler, M. McDaniel, D. Rohrer, and R. Bjork, "Learning styles concepts and evidence," *Psychological Science in the Public Interest*, vol. 9, no. 3, pp. 105-119, 2008.

[10] P. Ramiro, J. Navarro, I. Menacho, and M. Aguilar, "Estilo cognitivo reflexividad-impulsividad en escolares con alto nivel intelectual," *Revista Latinoamericana de Psicologia*, vol. 42, no. 2, pp. 193-202, 2010.

[11] C. Bégin, "Les stratégies d'apprentissage: Un cadre de référence simplifié," *Revue des sciences de l'éducation*, vol. 34, no. 1, p. 47–67, 2008.

[12] N. Adams, "Bloom's taxonomy of cognitive learning objectives," *Journal of the Medical Library Association*, vol. 103, no. 3, pp. 152-153, 2015.

[13] A. Crowe, C. Dirks, and M. Wenderoth, "Biology in bloom: Implementing Bloom's taxonomy to enhance student learning in biology," *CBE Life Sciences Education*, vol. 7, no. 4, pp. 368-381, 2008.

[14] N. Athanassiou, J. Mcnett, and C. Harvey, "Critical thinking in the management classroom: Bloom's taxonomy as a learning tool," *Journal of Management Education*, vol. 27, no. 5, pp. 555-575, 2003.

[15] W. Hanna, "The New Bloom's Taxonomy: Implications for Music Education," *Arts Education Policy Review*, vol. 108, no. 4, pp. 7-16, 2007.

[16] S. Masapanta-Carrión and J. Velázquez-Iturbide, "A systematic review of the use of bloom's taxonomy in computer science education," presented at 49th ACM Technical Symposium, 2018.

[17] M. V. Doran and D. D. Langan, "A cognitive-based approach to introductory computer science courses: lesson learned," presented at SIGCSE Technical Symposium on Computer Science Education, 1995.

[18] P. Machanick, "Experience of applying bloom's taxonomy in three courses," presented at Southern African Computer Lecturers' Association Conference, 2000.

[19] C. Johnson and U. Fuller, "Is Bloom's taxonomy appropriate for computer science?" presented at Baltic Sea Conference on Computing Education Research: Koli Calling, 2006.

[20] I. Sanders and C. Mueller, "A fundamentals-based curriculum for first year computer science," *ACM SIGCSE Bulletin*, 2000.

[21] E. Lahtinen, "A categorization of novice programmers: A cluster analysis study," presented at Workshop of the Psychology of Programming Interest Group, 2007.

[22] C. Starr, B. Manaris, and R. Stalvey, "Bloom's taxonomy revisited: Specifying assessable learning objectives in computer science," presented at ACM Technical Symposium on Computer Science Education, 2008.

[23] D. Oliver, T. Dobele, M. Greber, and T. Roberts, "This course has a bloom rating of 3.9," presented at Australasian Conference on Computing Education , 2004.

[24] E. Shneider and O. Gladkikh, "Designing questioning strategies for information technology courses," presented at 19th Annual Conference of the National Advisory Committee on Computing Qualifications (NACGQ), 2006.

[25] R. Lister and J. Leaney, "Introductory programming, criterion-referencing, and bloom," presented at SIGCSE Technical Symposium on Computer Science Education, 2003.

[26] R. Howard *et al*., "Felder's learning styles, bloom's taxonomy, and the Kolb learning cycle: Tying it all together in the CS2 course," presented at SIGCSE Technical Symposium on Computer Science Education, 1996.

[27] E. Thompson, A. Luxton-Reilly, J. Whalley, M. Hu, and P. Robbins, "Bloom's taxonomy for CS assessment," presented at Conference on Australasian computing education , 2008.

[28] T. Scott, "Bloom's taxonomy applied to testing in computer science classes," *Journal of Computing Sciences in Colleges*, vol. 19, no. 1, 2004.

[29] J. Whalley, R. Lister, E. Thompson, T. Clear, P. Robbins, P. Kumar, and C. Prasad, "An Australasian study of reading and comprehension skills in novice programmers, using the bloom and SOLO taxonomies," presented at Australasian Conference on Computing Education, 2006.

[30] E. Lahtinen and T. Ahoniemi, "Visualizations to support programming on different levels of cognitive development," presented at Fifth Koli Calling Conference on Computer Science Education, 2005.

[31] D. Cukierman and D. McGee-Thompson, "Learning strategies sessions within the classroom in computing science university courses," presented at Western Canadian Conference on Computing Education, 2007.

[32] U. Fuller, T. Johnson. C: Ahoniemi, D. Cukierma, I. Hernán-losada, J. Jackova, E. Lahtinen, T. Lewis, and C. T. E. Riedesell, "Developing a computer science-specific learning taxonomy," presented at Working Group Reports on ITiCSE on Innovation and Technology in Computer Science Education, 2007.

[33] Association for Computing Machinery. (2020). *Curricula Recommendations*. [Online]. Available: https://www.acm.org/education/curricula-recommendations

Sóniar Rolland Sobral was born in Porto in 1971. She is a professor at Universidade Portucalense since 1993 and currently she is a researcher at REMIT, research in economics, management, and information technologies. She is an aggregate (Dr. Habil) in information sciences, the doctorate (PhD) in information systems and technologies, the master (MSc) in electrical and computer engineering and degree in management informatics. She has more than 100 scientific publications and her focus is the distance education, serious games, computer programming and higher education policies. She is addicted to sports, and seriously passionate about technology and travel.