Received: 11 May 2023 | Revised: 3 October 2023 | Accepted: 22 January 2024

DOI: 10.1002/cae.22723

RESEARCH ARTICLE

WILEY

# Training future engineers: Integrating Computational Thinking and effective learning methodologies into education

**Rafael Herrero-Álvarez** | **Coromoto León** | **Gara Miranda** | **Eduardo Segredo**

Departamento de Ingeniería Informática y de Sistemas, Universidad de La Laguna, San Cristóbal de La Laguna, Santa Cruz de Tenerife, Spain

**Correspondence**
Rafael Herrero-Álvarez, Departamento de Ingeniería Informática y de Sistemas, Universidad de La Laguna, Camino San Francisco de Paula, n19, Módulo A of Escuela Superior de Ingeniería y Tecnología, San Cristóbal de La Laguna 38200, Santa Cruz de Tenerife, Spain.
Email: rherrero@ull.edu.es

**Funding information**
Cabildo de Tenerife and the Fundación General de la Universidad de La Laguna, Grant/Award Number: REF23120270; Fundación LaCaixa-Cajacanarias, Grant/Award Number: 2020EDU01; Canary Islands Agency for Research, Innovation and the Information Society of the Ministry of Universities, Science, Innovation and Culture and by the European Social Fund Plus (ESF+) Integrated Operational Program

**Abstract**

This article examines the effectiveness and interest generated among primary and secondary education students through activities aimed at developing Computational Thinking skills, in the context of the coronavirus disease 2019 pandemic. The shift to online or hybrid learning models posed a significant challenge for educators, particularly those lacking digital skills. The study sought to answer several research questions, including the impact of online versus in-person teaching on preuniversity students and gender differences in Computer Science perception, and Computational Thinking skills performance. The study employed a four-phase methodology, consisting of pre- and posttraining measurements of Computer Science perception and Computational Thinking skills development through specific activities delivered in-person or online. The results indicate that in-person training is more effective for developing Computational Thinking skills, particularly at the secondary education level. Furthermore, there is a need to focus on maintaining girls' interest in Computer Science during primary school, as interest levels tend to decline significantly in secondary school. These findings have significant implications for Engineering Education in the context of digital transformation and the increasing importance of Computational Thinking skills in various fields of engineering. This study highlights the importance of developing Computational Thinking skills among preuniversity students and the need for effective training methods to achieve this goal and underscore the significance of investing in Engineering Education to prepare the next generation of engineers for the rapidly changing digital landscape.

**KEYWORDS**

Computational Thinking, Computer Science, Engineering Education, primary education, secondary education

## 1 | INTRODUCTION

The coronavirus disease 2019 (COVID-19) pandemic has brought about significant changes across various sectors [18], including education. One of these changes is the shift toward distance learning, particularly the adoption of online learning or *e-learning* models that rely on digital devices and tools to support the learning process [2]. This transformation has disrupted the traditional educational models with which we are familiar [24] and has required teachers to adapt a wide range of resources and materials to the online environment while implementing new teaching methodologies [6, 14, 46].

This shift in teaching/learning methods has posed a challenge for educators. It is not just about mastering new digital tools and skills but also about recognizing that not all subjects lend themselves equally to different teaching approaches. While Computer Science, the study of computers and algorithmic processes, including their principles, their hardware and software designs, their applications, and their impact on society [47], might seem well-suited for this adaptation, the absence of fundamental Computer Science concepts in educational curricula hinders both teachers and students in this transition. Also, this situation reveals that users of online learning platforms for middle school students often avoid engaging with the intricacies of the learning process and making realistic choices regarding their educational path [8].

Given the growth of Computer Science in recent decades, it is necessary to educate students in this field by training them in these types of skills and preparing them for the digital world in which we live, since future citizens need to be able to not only use these types of tools but also be capable of choosing the most suitable ones and employing them correctly [23, 48]. Given this, it is strange that this field is not formally rooted in all preuniversity education cycles. Currently, training in this area focuses basically on the use of computers, known as digital literacy, and not on understanding how they work to learn how to program them [50]. Some of the reasons for this absence of Computer Science from the educational curriculum, from the perspective of administrators and teachers, is the potential difficulty managing the numerous agents involved [1], the problems training teachers, the teachers' lack of knowledge in this subject and the lack of consistency between educational agents, as happens in some countries where it is taught as a stand-alone subject, and in others where it is an interdisciplinary subject [43, 40, 49]. From the point of view of young people, this is a subject that does not arouse great interest, either because of their lack of knowledge or because they perceive it as too complex and beyond their reach [16, 25].

Some authors argue that an effective way to introduce Computer Science in preuniversity teaching is through Computational Thinking [3, 21], since specific activities improve perception about programming and Computer Science [22, 30, 41]. Furthermore, Computational Thinking can also foster interest in Science, Technology, Engineering, Arts, and Mathematics fields [12]. Computational Thinking is defined as the ability to understand and solve problems using concepts from Computer Science [51]. Other authors believe that the concept of programming should also be included due to the skills that are developed in students as they learn to program (e.g., reuse, mix different projects, etc.) and the change in their perspective of the world around them (e.g., by questioning ideas, establishing contact with other people, etc.) [7]. This would foster a different kind of thinking: Computational Thinking.

The general ignorance that young people have about this subject is reflected in the decreasing interest in degrees related to Computer Science [5]. Some papers show that training in Computational Thinking skills improve the perception of Computer Science in both genders, but there are significant differences between boys and girls, in addition to a certain disinterest that increases with age in the case of girls [22]. Therefore, it is important to keep in mind the concept of gender to keep girls interested in the subject [52].

There are initiatives that seek to bring this entire field closer to preuniversity studies, such as CODE.org with its hour of code project [10], or Google with CS First [17], which seeks to facilitate the teaching of programming and make it fun to learn. There are also initiatives with unplugged exercises, meaning they do not require the use of a computer or tablet, such as CS Unplugged [11]. Moreover, there are applications and tools such as Scratch [15, 36] that make it possible to program freely using a block-based visual programming language, and Microsoft's Arcade Makecode [33], which lets users create a video game using blocks or fragments of code in JavaScript or Python languages. There are also proposals that address creative thinking for preschool children [53], a skill deemed essential in the 21st century along with Computational Thinking [26]. Proposals have even emerged based on a service-learning model, such as the organization of a Computational Thinking Olympiad [19].

This paper studies whether there is a change in how preuniversity students perceive Computer Science if the subject is introduced through in-person or online activities. Several authors suggest that an online modality could be equally effective as in-person since it has not been possible to find significant differences that determine which modality is better for students [44]. Additionally, more efforts need to be applied to evaluation and the development of practices in this modality, especially in the Science, Technology, Engineering, and Mathematics (STEM) field [13]. The main hypothesis is that both the students' perception and their computational skills improve independently of the model used to present the activities. This research is being conducted as part of the Piensa ComputacionULLmente [20] project, which, since the 2017–2018 academic year, has allowed preuniversity students to carry out Computational Thinking activities in-person, but which during the 2020–2021 academic year had to be carried out online using *synchronous interactive teaching* [35] due to the COVID-19 pandemic.

The remainder of this paper is structured as follows: Section 2 presents the objectives, before continuing with the methodology used in Section 3. Section 4 presents and discusses the results obtained. Finally, Section 5 sets out the conclusions and lines of future work.

## 2 | GOALS AND HYPOTHESES

This study is being conducted as part of a project whose objective is to introduce Computer Science to preuniversity students by developing their Computer Thinking skills [20] so that they can understand what Computer Science is, and thus foster their interest in this subject. In addition, special attention is paid to girls, since the numbers of female students enrolled in engineering degrees is considerably low [45]. Given this scenario, another goal is for this type of activity to foster girls' interest in Computer Science [42].

The main hypothesis proposed in this work is that the "*perception*" of Computer Science improves in preuniversity students when they receive online training in activities related to the development of Computational Thinking skills, as happens when this training is provided in-person [22]. Therefore, we expect no significant differences between the two teaching/learning models. In this regard, the specific hypotheses proposed are as follows:

**H1.** The perception of Computer Science that preuniversity students have is improved through specific training carried out in-person or in synchronous interactive teaching.

**H2.** There are no significant differences depending on whether the training is carried out in-person or online.

Another hypothesis we propose is that there are also no significant differences in the development of Computational Thinking "*skills*" between the two training models, meaning that after specific training, not only will the students' perception of Computer Science improve, but so will their skills or abilities related to Computational Thinking. Furthermore, no differences in the results between girls and boys are expected either. Taking the above into account, the specific hypotheses proposed are as follows:

**H3.** Girls and boys have a similar perception of Computer Science.

**H4.** Computational Thinking skills are improved after completing the proposed training.

**H5.** The Computational Thinking skills or performance of students is independent of gender.

Table 1 summarizes the relationship between the goals and hypotheses proposed in this work.

## 3 | METHODOLOGY

The methodology followed focuses on the two main variables of this study: the perception of Computer Science and Computational Thinking skills. Since these are different constructs, to measure each of them in the students, we will need to use a different instrument. In addition, to assess the impact of the training we propose, we must define a control group and an experimental group. To make comparisons between an online methodology and one that is entirely in-person, we will need to compare the results obtained before the pandemic with those obtained when the training provided had to be adapted due to the pandemic. Based on these methodological issues, the following phases were defined:

**Phase 1.** Measurement of the students' perception of Computer Science before receiving the training (*pretest*: see Section 3.1).

**Phase 2.** Training for the students involving a series of specific activities intended to develop their Computational Thinking skills. This training can be provided in-person or online (Training: see Section 3.2).

**Phase 3.** Measurement of the students' perception of Computer Science after receiving the training (*posttest*: see Section 3.1).

**Phase 4.** Measurement of the level of the Computational Thinking skills (Computational Thinking Test [CTT]: see Section 3.3).

TABLE 1 Relationship between the goals and the hypotheses.

| Goals | Hypothesis |
|---|---|
| Introduce Computer Science to preuniversity students by developing their Computer Thinking skills so they can understand what Computer Science is | **H1.** The perception of Computer Science that preuniversity students have is improved through specific training provided in-person or in synchronous interactive teaching |
| | **H2.** There are no significant differences depending on whether the training is carried out in-person or online |
| Foster girls' interest in Computer Science | **H3.** Girls and boys have a similar perception of Computer Science |
| | **H4.** Computational Thinking skills are improved after completing the proposed training |
| | **H5.** The Computational Thinking skills or performance of students are independent of gender |

| | Pretest | Posttest |
|---|---|---|
| **I1.** How much do you like Computer Science? | ✓ | ✓ |
| **I2.** How much do you know about Computer Science? | ✓ | ✓ |
| **I3.** Do you think Computer Science is hard or difficult to learn? | ✓ | ✓ |
| **I4.** Do you think Computer Science is important? | ✓ | ✓ |
| **I5.** How much do you think you need to learn about Computer Science? | ✓ | ✓ |
| **I6.** Did you like the Computational Thinking activities that were presented? | | ✓ |

TABLE 2 Questions from the test to measure perception of Computer Science.

## 3.1 | Test to measure the perception of Computer Science

To confirm the perception that students have about Computer Science, a questionnaire was designed using a Likert [27] scale for the students to complete twice, once in Phase 1 before taking part in the activities (called *pretest*), and another later, in Phase 3 (called *posttest*), to study the interest that a given training activity aroused in students.

The instrument consists of six questions that can be answered on a numerical scale from 1 to 5, where 1 indicates that they do not like or have no knowledge about what is being asked, and 5, which indicates that they like or do have knowledge about what is being asked. The last question, number 6, is only answered in Phase 3 (*posttest*). The questions that comprise this test are listed in Table 2.

## 3.2 | Computational Thinking training

In Phase 2, two models were used to carry out the activities: one in-person (see Section 3.2.1) and another online (see Section 3.2.2). The training was provided in-person in the 2018–2019 school year, and online in 2020–2021. In both cases, five 2-h sessions were held with primary school students aged 8–9, and with secondary school students aged 12–13. The researcher responsible for the training went to the schools in the 2018–2019 school year to carry out the activities. In the 2020–2021 academic year, the researcher responsible for the training gave the sessions by videoconference, using synchronous interactive teaching, in the classroom where the students were, together with the teacher from the school. Note that the same person led all the training activities, in both modalities. The training consisted of plugged and unplugged activities. The plugged activities relied on computers or tablets, and the unplugged activities used materials such as pens and printed cards. The goal of these activities was to have the students work with Computational Thinking concepts, such as decomposition, pattern recognition, and abstraction.

The activities have been designed to be as similar as possible between both modalities, but due to logistical reasons, it has not been feasible to carry out the exact same activities.

### 3.2.1 | In-person training

The activities that were carried out during the 2018–2019 school year were presented in-person. They were divided by educational level (primary and secondary). Likewise, they were designed following two methodologies, one guided and one through discovery. In the case of the guided methodology, the concepts related to Computational Thinking were introduced using an example that was solved step by step. Once the example was done, the students continued to do exercises with instructions. As for the methodology through discovery, a tool was briefly introduced that explained the different options available, after which the various exercises were presented to the students for them to solve independently using this tool.

In primary school, the students in the guided methodology programmed the *Code&Go* robot [29], a mouse that has to cross a maze and that has to be programmed using the buttons at the top. They also took a course, Course 2, on the platform CODE.org [9], while the students in the discovery methodology designed and built a guitar using cards and aluminum foil, which they connected to the *Makey Makey* board [28]. They then implemented a program in Scratch [36], in addition to simulating the game of *Pong* on this same platform. The full distribution of activities by session is listed in Table 3.

In secondary school, students in the guided methodology took the Accelerated Intro to CS Course on the CODE.org platform, and they implemented the *Pong* game following instructions. Students in the discovery methodology worked with the *mBot* [32], a robot that has different sensors and two motors. The students were asked to implement a program whereby the robot was able to travel along a course independently using the *mBlock* [32] program. This course had to be designed by the students using white cardboard on which they drew the course with black markers, since the robot is able to distinguish if it is on a black or white background. It can also use an ultrasound sensor to detect possible obstacles. The *Pong* game was also implemented in this methodology, but autonomously. The full distribution of activities by session is listed in Table 4.

The unplugged activities were designed following the principles outlined by Looi et al. [31] to foster Computational Thinking by emphasizing connections between concepts, promoting analogical reasoning, and highlighting the potential for incorrect analogies. In the case of the plugged activities, consideration was given to the use of physical devices, which can actively engage students in problem-solving and facilitate the acquisition of powerful ideas from computer science and robotics, including core concepts of Computational Thinking [4].

However, it is important to note that it has been demonstrated that there are no significant differences between using guided and discovery methodologies, either in primary or secondary school [22], so it was decided to discard this approach for the 2020–2021 academic year.

### 3.2.2 | Online training

The activities done during the 2020–2021 academic year were carried out online using synchronous interactive

**TABLE 3** Description of the primary education activities performed in-person.

| First session | | Second session | Third session | Fourth session | Fifth session | |
|---|---|---|---|---|---|---|
| Guided methodology | | | | | | |
| Pretest CTT | *Code&Go* robot | CODE.org (Maze) | CODE.org (Artist) | CODE.org (Bee) | CODE.org (Loops, debugging) | Posttest CTT |
| Discovery methodology | | | | | | |
| Pretest CTT | *Makey Makey* demo | Design and build a guitar | Pong game in Scratch | Program guitar in Scratch | Program guitar in Scratch | Posttest CTT |

**TABLE 4** Description of the secondary education activities performed in-person.

| First session | | Second session | Third session | Fourth session | Fifth session | |
|---|---|---|---|---|---|---|
| Guided methodology | | | | | | |
| Pretest CTT | CODE.org (Maze) | CODE.org (Artist) | CODE.org (Farmer) | Guided Scratch (PONG game) | CODE.org (Functions, conditionals) | Posttest CTT |
| Discovery methodology | | | | | | |
| Pretest CTT | mBot demo | Design and build a circuit | PONG game in Scratch | PONG game in Scratch Program circuit in mBlock | Program and test mBot | Posttest CTT |

teaching and a videoconference system, with the students in schools with their corresponding teacher, and the researcher in charge of the project in a studio. The activities were designed so that the first hour of the session involved unplugged exercises, without using either computers or tablets, and the second hour involved plugged exercises, since this combination motivates students compared to just one kind of activity [34]. Each session practiced a different programming concept; specifically, sequences, conditionals, loops, variables and functions.

These activities aimed to follow the same structure as those designed for in-person training, considering that it might be more difficult for schools to access physical devices such as the *Code&Go* robot or the *Makey Makey* board.

For the primary school students, the first session presented the "Cross the maze" exercise using sequences, as well as sections 1–4 of Course 2 of CODE.org. In the second session, the students worked on conditionals with the "Decision tree" exercise, where they have to find the number that a classmate was thinking about, and with the conditionals in Course D of CODE.org. In the third session, the students completed the activity "Programming a drawing," which had them draw on a grid as per the instructions given. Also, given a drawing, they had to determine the necessary instructions using loops. Finally, they had to do an exercise using Scratch, which is programmed to draw a square, a triangle and a circle. The fourth session practiced variables with the "Variable envelopes" exercise, where the goal is to draw a monster after receiving values in different envelopes containing characteristics of the monster. They also had to do an activity in Scratch that consisted of exploding balloons by moving a pencil. In the final session, students worked with the "My own remote control" functions, which shows some buttons on a sheet of paper with a geometric figure, each one containing a fragment, and the students have to form a complete sentence such as a song or a riddle. They also had to program the game *Pong* in Scratch. The full distribution of activities by session is listed in Table 5.

For the secondary school students, the first session involved the activity "Castles of glasses," where they had to assemble a castle of glasses following a sequence of arrows. They also did sections 1–4 of Course 2 of CODE.org. In the second session, they were presented a "Sorting" activity that required them to sort numbers using a pairwise sorting algorithm, in addition to sections 10–13 of Course D (2017) of CODE.org. In the third session, the students worked on loops using the same activity as in the first session, "Castle of glasses," but using loops. They also worked on sections 5–8 of Course 2 of CODE.org. In the fourth session, they worked on variables with the exercise "Maze with variables," where they had to navigate through a maze with different variables whose values were updated during the activity. They also had to program the *Pong* game in Scratch. In the last session, to practice functions, they did the "Hero or villain" activity in which one group had to destroy a city and others had to defend it by using cards with different attack and defense points. They also did an activity in Scratch where a cat had to catch different mice that appeared on the screen. The full distribution of activities by session is listed in Table 6.

## 3.3 | CTT

To verify the effectiveness of the activities, experimental and control groups were set up in Phase 1. In this case, a published and validated instrument was used (see Section 3.3) The Computational Thinking skills were measured using version 2.0 of the CTT from November 2014 [37, 39]. This instrument is designed for students up to 16 years of age and consists of a total of 28 multiple-choice questions, all related to different programming concepts, which have to be solved using a block-based visual programming language. Students have a total of 40 min to complete this test. The authors propose classifying the test questions based on the following concepts: movements in four directions, basic directions (four specific items), finite loops or loops that are

**TABLE 5** Description of the primary education activities performed online.

| First session | | Second session | Third session | Fourth session | Fifth session | |
|---|---|---|---|---|---|---|
| Unplugged activities | | | | | | |
| Pretest CTT | Walking the maze | Decision tree | Programming a drawing | Variables envelopes | My remote control | Posttest CTT |
| Plugged activities | | | | | | |
| Pretest CTT | CODE.org (Sequences) | CODE.org (Conditionals) | Figures in Scratch | Popping balloons | PONG game in Scratch | Posttest CTT |

**TABLE 6** Description of the secondary education activities performed online.

| First session | | Second session | Third session | Fourth session | Fifth session | |
|---|---|---|---|---|---|---|
| Unplugged activities | | | | | | |
| Pretest CTT | Cups castles (Sequences) | Ordered numbers | Cups castles (Loops) | Maze with variables | Hero or villain | Posttest CTT |
| Plugged activities | | | | | | |
| Pretest CTT | CODE.org (Sequences) | CODE.org (Conditionals) | CODE.org (Loops) | PONG game in Scratch | Catching the mouse | Posttest CTT |

**TABLE 7** Quantitative description of the student sample: in-person for the 2018–2019 academic year, and online for 2020–2021.

| Primary education | | | | Secondary education | | | |
|---|---|---|---|---|---|---|---|
| In-person 4 schools 126 students | | Online 3 schools 56 students | | In-person 6 schools 145 students | | Online 4 schools 116 students | |
| 70 | 56 | 33 | 23 | 64 | 81 | 56 | 60 |
| Girls | Boys | Girls | Boys | Girls | Boys | Girls | Boys |

repeated until a condition is met (12 items), conditional with a single requirement, compound conditionals, or that are executed while a condition is met (eight items), and functions (four items), as well as nesting. The students receive a score from 0 to 28 (each right answer is worth one point).

This tool can be used for pretest assessments to gauge the initial CT development of students with no prior programming experience. It can also be employed collectively for large-scale screenings, early identification of students with programming aptitude or special needs, and to gather quantitative data for pre–post evaluations of CT-focused curricula or programs. This quantitative approach complements the predominantly qualitative methods used in the existing literature and can be integrated into academic and professional STEM guidance processes [38].

To properly check how this training affects Computational Thinking skills, two groups of students in Phase 4 took this questionnaire. One was the control group, which did not receive the training, and another was the experimental group, which did receive it. The two groups were distributed using logistical reasons involving the participating schools. The students were all in the same educational level, the same age group, and some even went to the same school.

## 3.4 | Sample and data analysis

The project was carried out in 15 different primary and secondary schools, 13 public, one private-aided, and one private, involving a total of 443 students, 182 of whom were in primary education, between 8 and 9 years old, and 261 in secondary education, between 12 and 13 years old. Two of the secondary schools participated in both editions, but with different students.

Table 7 shows the complete data based on the ages, sex, and program followed by the students, based on the academic year in which they received the training.

Before analyzing the data, it was preprocessed to eliminate duplicates and discordant entries between the *pretest* and *posttest*. This implies that the number of students in the sample does not match the number of data points in the perception study or in the skills study.

Once the results were preprocessed, two different analyses were carried out. On the one hand, the data from the *pretest* and the *posttest*, which measure the students' perception of Computer Science, was used to study the existence of significant differences between the in-person and online models using the Student's $t$-distribution with a significance level of 95% ($p < .05$), to accept or reject the hypotheses proposed. The average score for each question and for each model were also calculated. On the other hand, to check the level of the students' Computational Thinking skills, the Kolmogorov–Smirnov test was performed to check the normality of the sample. The Student's $t$-distribution study with a significance level of 95% ($p < .05$) was also carried out to check for the possible existence of significant differences. Finally, Pearson's test was performed to study the correlation between certain questions and their answers. The average score obtained by the students was taken into account to check the variations between the two models.

## 4 | RESULTS AND DISCUSSION

This section presents the results obtained for the tests administered in Phase 1 (*pretest*) and Phase 3 (*posttest*). The results obtained in Phase 4 for the CTT are also analyzed. The results are shown taking into account the model (in-person vs. online), the educational level and gender.

To assess the existence of significant differences in the results, all of these were examined using the Student's *t*-distribution with a significance level of 95% ($p < .05$). Additionally, for the CTT, a Kolmogorov–Smirnov test was conducted to check for normality.

### 4.1 | Perception of Computer Science

To determine the students' perception of Computer Science, the answers given to the *pretest* in Phase 1, and to the *posttest* in Phase 3, were studied. Before the study, the *pretests* were analyzed using the Student's *t*-distribution with a significance level of 95% ($p < .05$), which revealed that there were no significant differences between the in-person and online samples obtained before the training. However, when the differences between the answers to the *post-tests* were analyzed, we found significant differences in some of the questions, both in primary and secondary education, causing us to reject hypothesis H2 (*There are no significant differences depending on whether the training is carried out in-person or online*). Specifically, in the primary school students, there are significant differences for every question, except for I5 (*How much do you think you need to learn about Computer Science?*), as shown in Table 8. In the secondary school students, there are only significant differences in question I2 (*How much do you know about Computer Science?*). A priori, it is interesting to consider that training on Computational Thinking can have a greater impact at earlier ages.

Analyzing the differences between boys and girls, in primary school there were no significant differences in the *pretest*, but there were in the *posttest*, specifically question I1 (*How much do you like Computer Science?*), as shown in Table 9. If we analyze the average scores for each question, as shown in Figures 1 and 2, the answers are similar in both cases, although the lower interest or affinity that girls exhibit for Computer Science (I1) when doing these activities online is notable, and is not evident when the activities are presented in-person. Also of note is the fact that girls view Computer Science as more difficult (I3) after receiving online training, while the boys' responses are similar in both cases. Despite this, the perception improves in primary education in both cases.

In secondary education, there are significant differences between boys and girls, both in the *pretest* and *posttest*, although these are apparent only for the in-person activities, as shown in Table 10. For the online training, no significant differences were found. When analyzing the average scores, as shown in Figures 3 and 4, regardless of the type of training, the score increases for questions I1 and I2, on whether they like Computer Science and how much they think they know, and decreases for those related to the difficulty (I3, I4) and how much they think they have to learn (I5).

As the analysis has shown so far, the perception of Computer Science improves for both educational levels when training involving Computational Thinking skills is provided, either in-person or online; thus, hypothesis H1 is accepted (*The perception of Computer Science that preuniversity students have is improved through specific training carried out in-person or in synchronous interactive teaching*). Regarding gender differences, girls and boys in primary school have a similar perception of Computer Science, so hypothesis H3 is accepted (*Girls and boys have a similar perception of Computer Science*) for primary education. However, there are significant differences in secondary education, since boys seem to exhibit more confidence in Computer Science than girls, so hypothesis H3 is rejected for secondary education.

**TABLE 8** *p* Values obtained when comparing the answers to the *posttest* for the in-person and online models.

| | I1 | I2 | I3 | I4 | I5 |
|---|---|---|---|---|---|
| Primary | **0.0002** | **0.0379** | **0.0174** | **0.0059** | 0.0740 |
| Secondary | 0.3418 | **0.0339** | 0.7005 | 0.8202 | 0.2644 |

*Note*: Statistically significant differences are shown in bold-face.

**TABLE 9** *p* Values obtained when comparing the answers to the *posttest* for the in-person and online models and gender in primary education.

| | I1 | I2 | I3 | I4 | I5 |
|---|---|---|---|---|---|
| In-person | **0.0121** | 0.6759 | 0.6626 | 0.0590 | 0.0970 |
| Online | **0.0306** | 0.2828 | 0.6852 | 0.7974 | 0.1123 |

*Note*: Statistically significant differences are shown in bold-face.

### 4.2 | Assessment of the training

This section presents the results of question I6 (*Did you like the Computational Thinking activities that were presented?*), which was only asked in the *posttest*, in Phase 3.

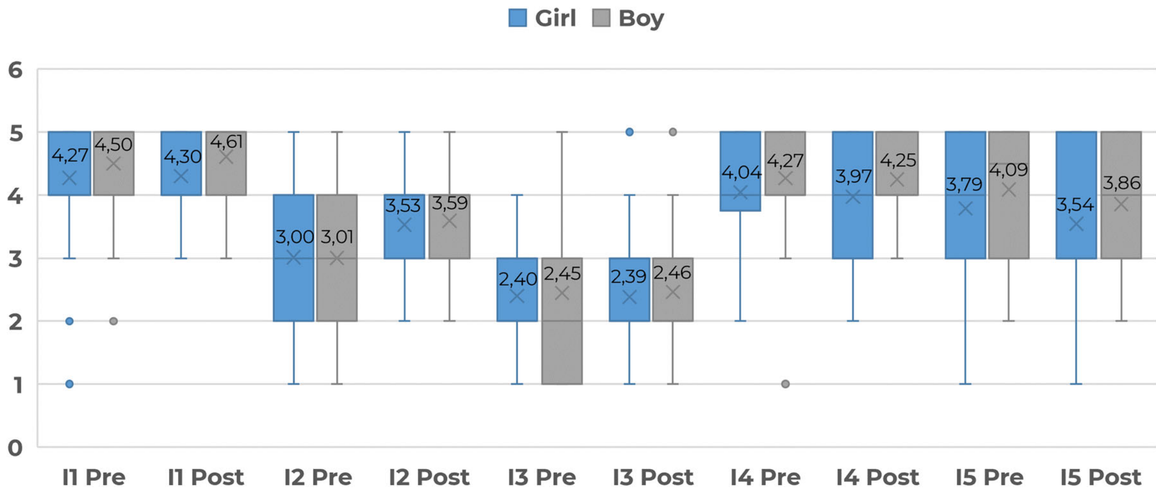


**FIGURE 1** Result of the *pretest* and *posttest* for primary education with in-person training, taking gender into account. A box plot is provided for quartiles Q1 and Q3, showing also the average score for each question.

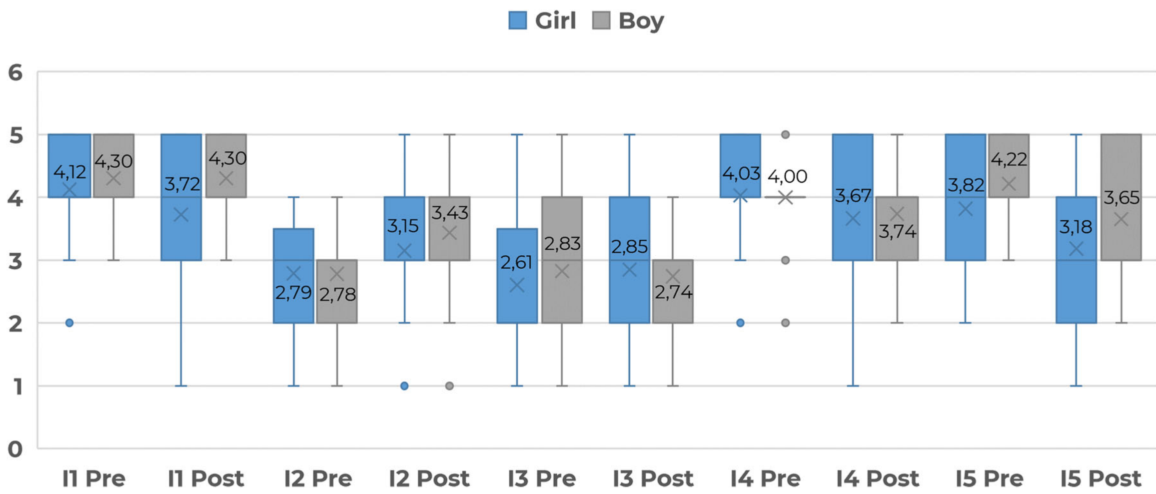


**FIGURE 2** Result of the *pretest* and *posttest* for primary education with online training, taking gender into account. A box plot is provided for quartiles $Q1$ and Q3, showing also the average score for each question.

**TABLE 10** $p$ Values obtained when comparing the answers to the *pretest* and *posttest* for the in-person model and gender in secondary education.

| | I1 | I2 | I3 | I4 | I5 |
|---|---|---|---|---|---|
| Pretest | **0.0005** | **0.0266** | 0.3236 | 0.2438 | 0.4757 |
| Posttest | **0.0003** | 0.6052 | 0.5526 | 0.7739 | **0.0004** |

*Note*: Statistically significant differences are shown in bold-face.

Figure 5 shows the mean scores for this question in primary education. For each stage, a distinction is made between in-person and online training, and between girls and boys. As the figure shows, most of the students liked this type of activity "more than other types of activities" (four out of five) or even "a lot" (five out of five). However, there is a significant difference between the in-person and online models, going from 88.27% to 73.97% of students, respectively. As Table 11 shows, the Student's $t$-distribution data with a significance level of 95% ($p < .05$) exhibits significant differences between the in-person and online training.

Apart from the significant difference seen in girls but not in boys, there is an observed disparity on average ratings between the two genders. Girls tend to rate the activities more negatively than boys, especially during online sessions.

Meanwhile, Figure 6 shows the average scores for this question in secondary education, also differentiating between in-person and online training, and gender. As in primary education, most of the students liked these activities "more than other types of activities" (four out of five) or even "a lot" (five out of five).

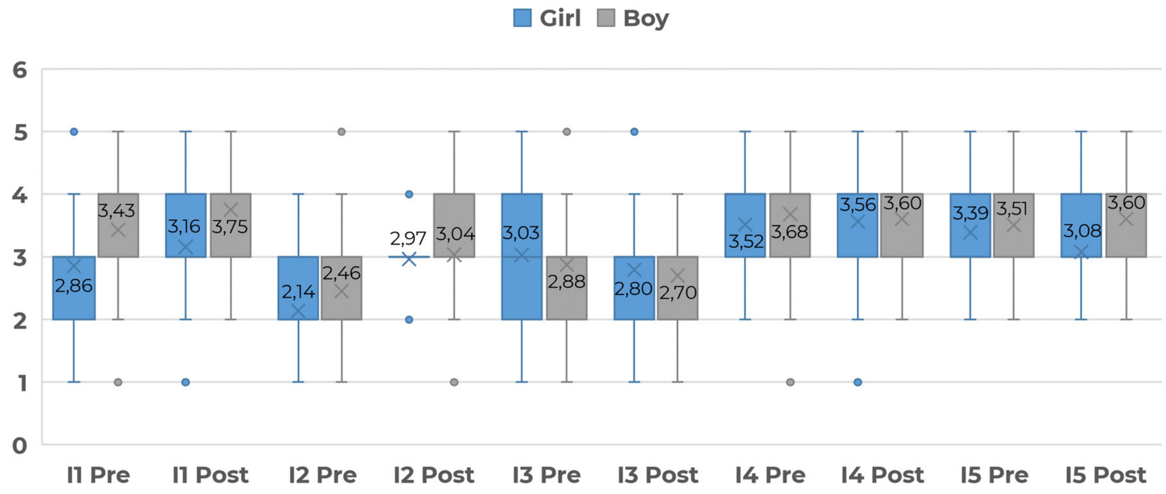


FIGURE 3 Result of the *pretest* and *posttest* for secondary education with in-person training, taking gender into account. A box plot is provided for quartiles $Q1$ and $Q3$, showing also the average score for each question.

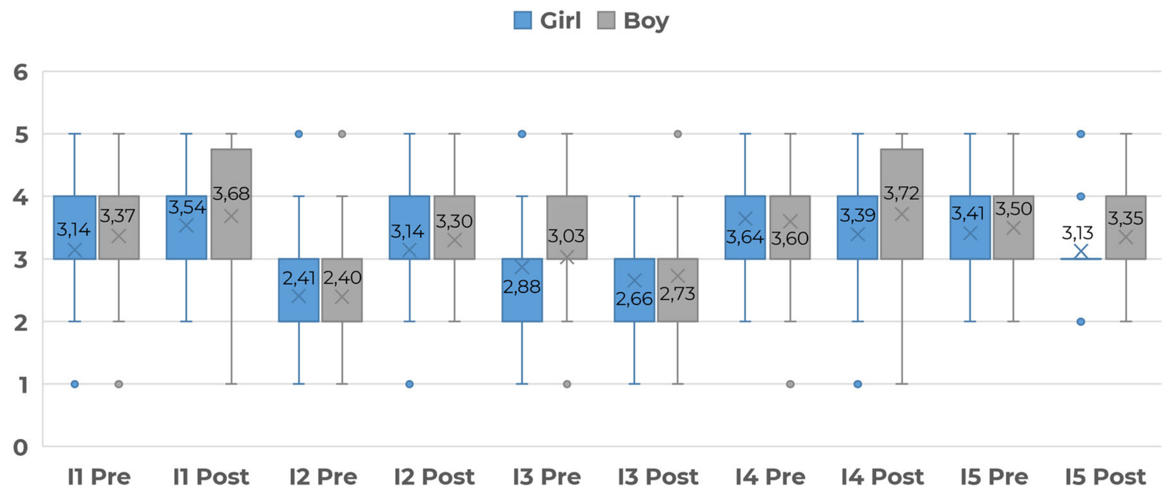


FIGURE 4 Result of the *pretest* and *posttest* for secondary education with online training, taking gender into account. A box plot is provided for quartiles $Q1$ and Q3, showing also the average score for each question.

However, this percentage decreases from 78.57% for in-person training to 67.20% for tonline training. As shown in Table 11, the age of the students affects the rate of activities, as the training model has an impact on primary school children that is not evident among secondary school children, for whom there are no differences based on the model.

Similar to what occurs in primary education, girls tend to evaluate activities more negatively than boys, although now significant differences are noticeable among boys depending on the model of activities.

Hypothesis H2 (*There are no significant differences depending on whether the training is carried out in-person or online*) is rejected for both primary and secondary education, since in both cases there are significant differences. The in-person training is preferred by students, regardless of their educational stage. In any case, the choice of training seems to have a greater impact on younger students.

## 4.3 | Efficacy of Computational Thinking training

To conduct this study, the students were divided into two groups—control and experimental—and the test was administered in Phase 4 for both the in-person and online training. Table 12 shows the sample based on the group that did the test. To study the normality of the sample, the Kolmogorov–Smirnov test was carried out, and as the $p$ values in Table 13 show, the normality was accepted for all samples.

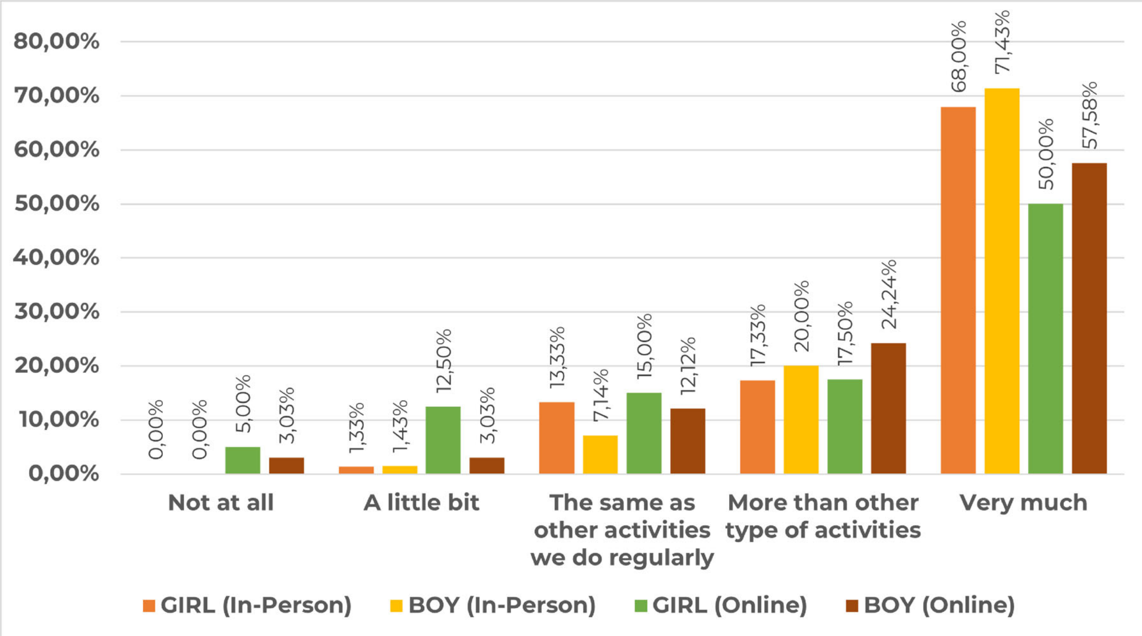


**FIGURE 5** Average scores for the question on the perception of the activities in primary education, distinguishing by model used and gender.

**TABLE 11** *p* Values obtained when comparing the answers in which the primary and secondary education students are asked to rate the activities in the in-person and online models.

| | Both | Girls | Boys |
|---|---|---|---|
| Primary | **0.0005** | **0.0036** | 0.0703 |
| Secondary | 0.1450 | 0.4591 | **0.0077** |

*Note*: Statistically significant differences are shown in bold-face.

Figure 7 shows the scores obtained in the CTT for primary education, for both the control and experimental groups, based on the training method used. The average score, out of a maximum of 28.0 points, was 10.2 for the girls and 9.7 for the boys in the control group, and 13.5 for the girls and 14.6 for the boys in the experimental group for the in-person training. For the online model, the average scores obtained were 9.1 for girls and 10.7 for boys in the control group, and 10.5 for girls and 11.7 for boys in the experimental group. A Student's *t*-distribution with a significance level equal to 95% ($p < .05$) was also calculated, taking into account all students, regardless of their gender, and a $p$ value of $3.48e-09$ was obtained for the in-person training and 0.1939 for online, yielding significant differences in the in-person option. Based on the average scores, hypothesis H4 (*Computational Thinking skills are improved after completing the proposed training*) is accepted for primary education and for the two models, since in both cases the value obtained in the test is higher.

Figure 8 collects the average scores of the secondary school students in the CTT, for both the control and experimental groups, differentiating by the training model. The average score in the control group was 15.6 for the girls and 16.2 for the boys, and 17.2 for the girls and 19.4 for the boys in the experimental group, for the in-person training. In the case of the online training, the average for the girls was 14.4 and 14.3 for the boys in the control group, and 13.3 for the girls and 12.6 for the boys in the experimental group. When the Student's *t*-distribution with a significance level equal to 95% ($p < .05$) was calculated, a $p$ value of 0.0008 was obtained for the in-person training and 0.2016 for the online modality. There is an improvement in the score for the in-person training, while for the online option, the score decreases; as a result, hypothesis H4 is rejected for secondary education.

Figure 9 shows the average scores obtained by the primary school students for each concept, separated by training model, control or experimental group, and gender. A positive overall change is evident in both girls and boys, showing higher scores in practically every concept, although this change is more significant for the in-person model. We can also see how the concepts relating to conditionals received the lowest score, for both girls and boys, and how the concept of directions has the highest score. In the case of loops, the score is also good for both loops until and loop times.

When determining the Student's *t*-distribution, in the case of the in-person training, the $p$ value obtained is

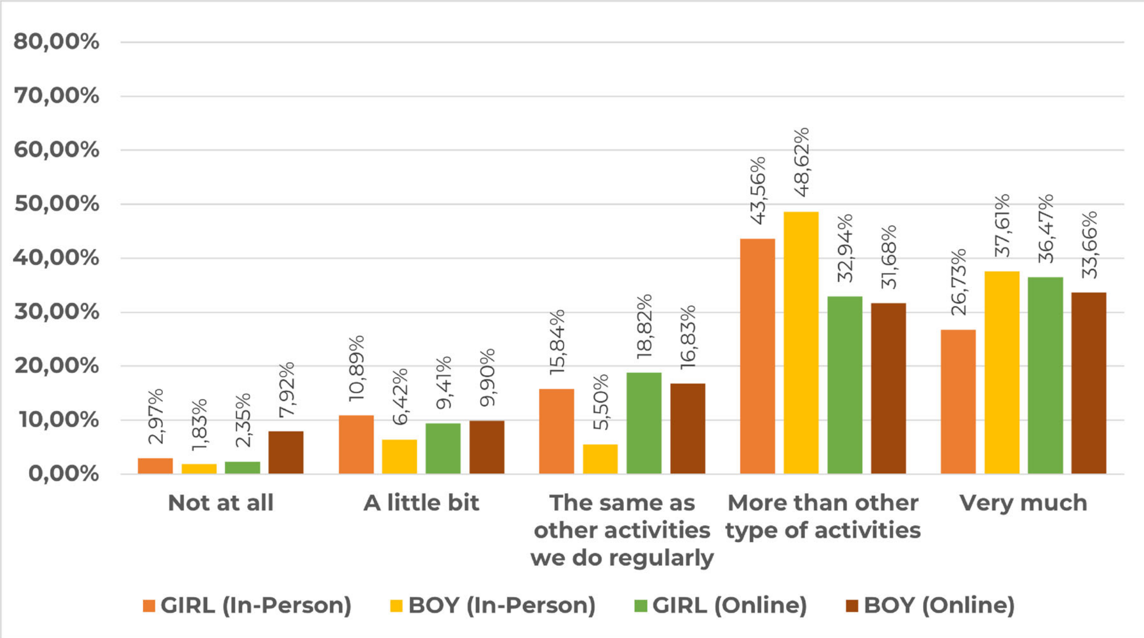


FIGURE 6 Average scores for the question on the perception of the activities in secondary education, distinguishing by model used and gender.

TABLE 12 Sample of students in the CTT.

| Primary education | | | | Secondary education | | | |
|---|---|---|---|---|---|---|---|
| In-person 141 students | | Online 66 students | | In-person 210 students | | Online 60 students | |
| Control | Experimental | Control | Experimental | Control | Experimental | Control | Experimental |
| 65 | 76 | 31 | 35 | 80 | 130 | 36 | 24 |
| Students | Students | Students | Students | Students | Students | Students | Students |

TABLE 13 *p* Values obtained for the normality of the sample using the Kolmogorov–Smirnov test.

| Primary education | | | | Secondary education | | | |
|---|---|---|---|---|---|---|---|
| In-person | | Online | | In-person | | Online | |
| Control | Experimental | Control | Experimental | Control | Experimental | Control | Experimental |
| 0.553 | 0.341 | 0.375 | 0.561 | 0.723 | 0.459 | 0.800 | 0.916 |

0.5526 and 0.1615 for the online model, so no significant gender differences are observed, thus validating hypothesis H5 (*The Computational Thinking skills or performance of students is independent of gender*) in primary education.

Figure 10 shows the average scores for secondary education, based on the different concepts, and the training model employed, and whether it was the control group or the experimental group. In the case of the in-person training, there is a positive change in both genders, although it is more pronounced in the case of boys. However, for the online model, in most of the concepts, the score is equal or lower after training for both girls and boys. By concept, the secondary school students excel in loops, and especially in directions, while questions about conditionals and functions yield the worst score. This happens with both girls and boys.

A determination of the Student's *t*-distribution yielded a *p* value 0.0354 for the in-person training and 0.7791 for the online training; thus, only significant gender differences are observed for the in-person training, meaning hypothesis H5 is rejected for secondary education.

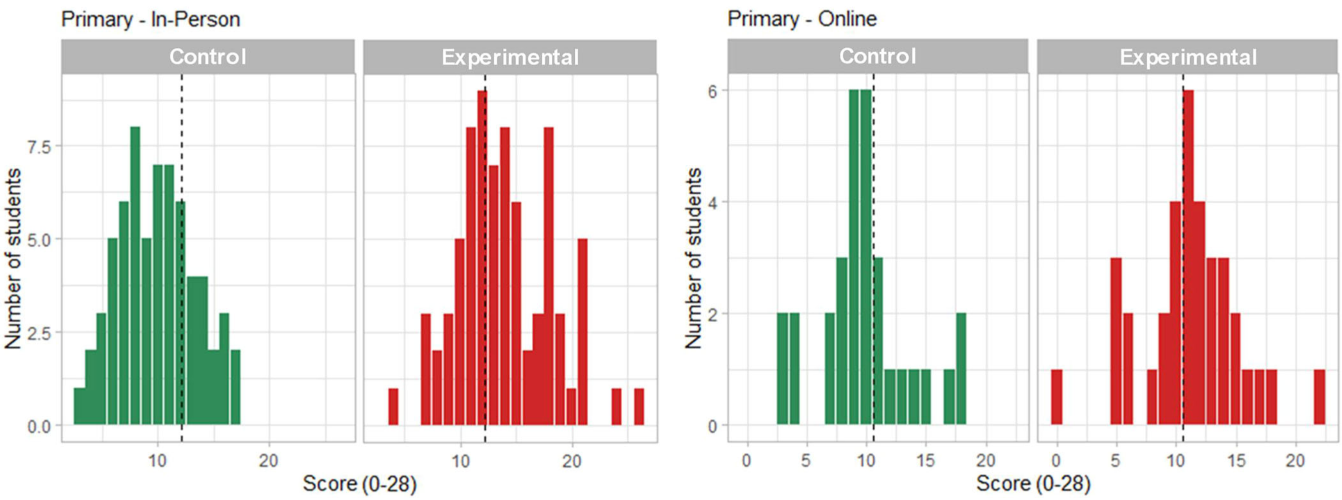


FIGURE 7 Result of the Computational Thinking Test for primary education, showing the average score by training type.

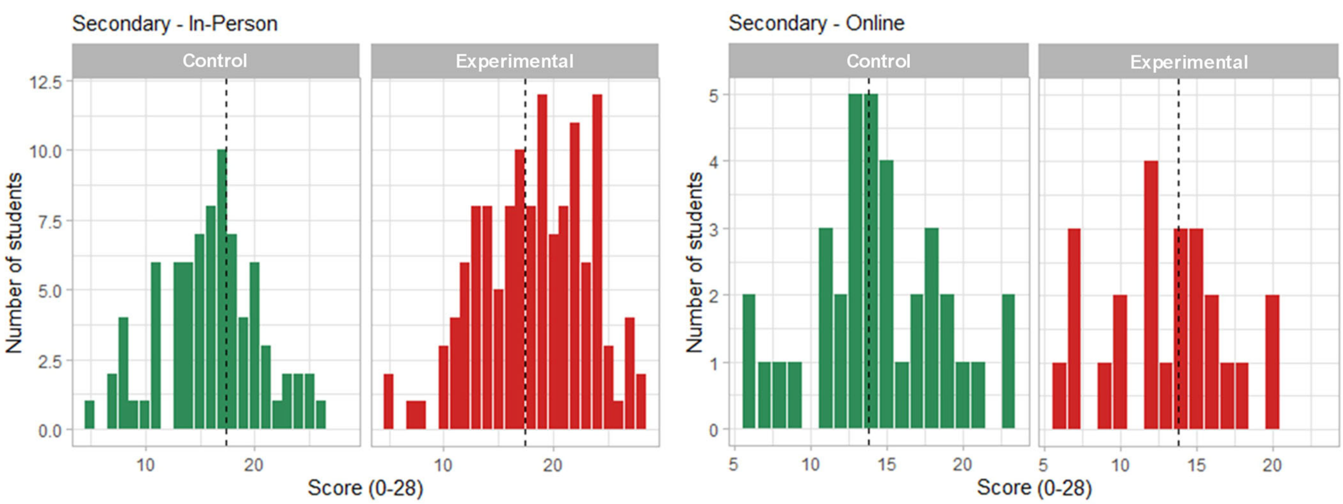


FIGURE 8 Result of the Computational Thinking Test for secondary education, showing the average score by training type.

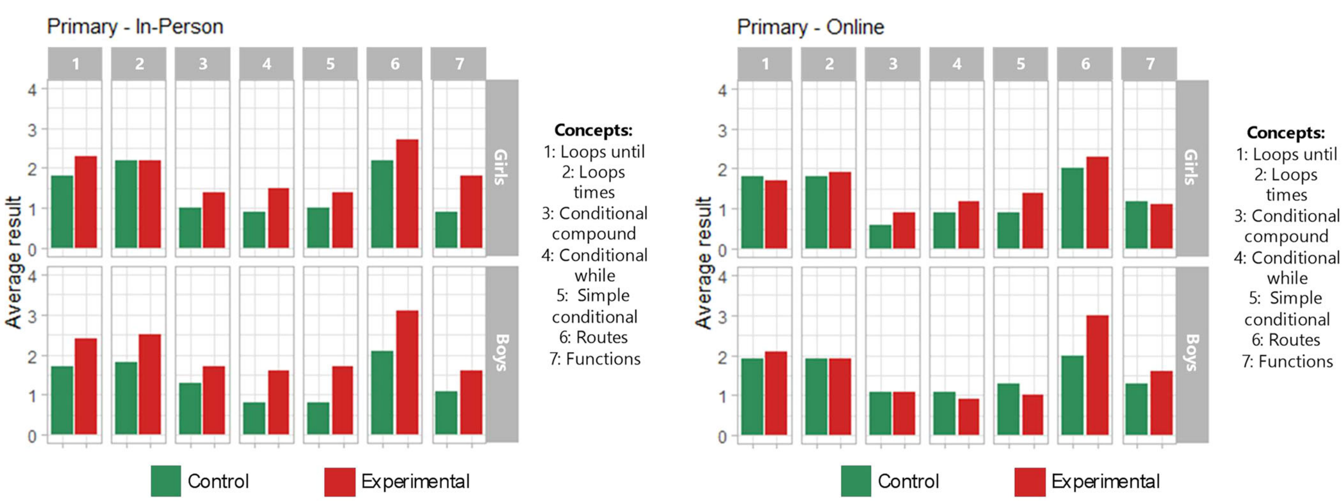


FIGURE 9 Average scores for the Computational Thinking Test for primary education based on the training received, distinguishing by results before and after the training, and also by gender.

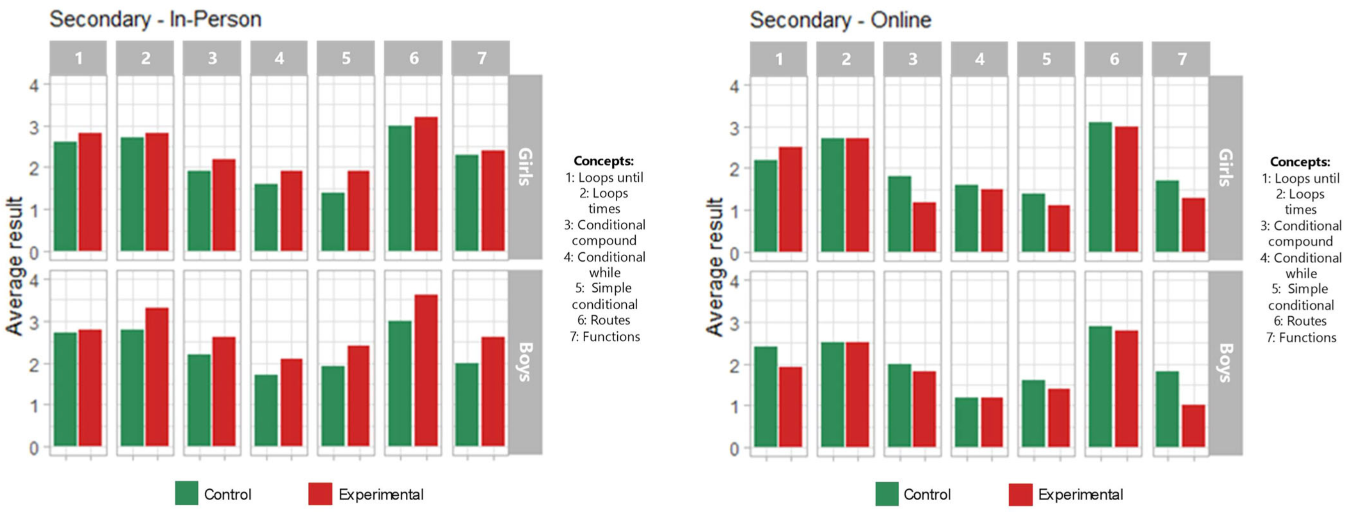


**FIGURE 10** Average scores for the Computational Thinking Test for secondary education based on the training received, distinguishing by results before and after the training, and also by gender.

## 5 | CONCLUSIONS

The main objective of this work is to compare two teaching/learning methodologies: fully classroom-based versus online. As a case study, this work has focused on specific training to develop Computational Thinking while improving the perception that preuniversity students have of Computer Science. This paper proposes a methodology for measuring the impact that different types of training have on Computational Thinking. We also conduct an in-depth analysis of the students based on their gender and age.

Based on our results, the perception of Computer Science improves independently of the training model. However, the level of improvement obtained is different for each model: both primary and secondary school students prefer in-person over online training. That is, training in Computational Thinking constitutes in and of itself a way to improve the overall perception of Computer Science; despite this, the methodological design as well as the type of activities carried out have a considerable impact on the improvement that can be achieved. It should be noted that improving the perception of this subject does not always imply an improvement in the interest or motivation that can be generated in the students. In fact, according to the results, online training could actually discourage students. Thus, our results indicate how the interest in the proposed activities decreases both in primary and secondary education when it is carried out using synchronous interactive teaching.

Regarding the Computational Thinking skills, in the primary school students, there is an improvement in both training types, but the difference is not as noticeable, with the improvement being greater when the activities are done in-person. In the case of secondary education, these skills improve if they are done in-person, but they worsen when the training is provided online, which reflects the lesser interest they exhibit in the activities if presented using this model.

This study has demonstrated the existence of significant gender differences in the perception of Computer Science, especially among girls during secondary education, a time when their interest decreases. This could explain the growing lack of interest in older girls. Furthermore, these differences are more pronounced when they engage in online activities, with some even considering this field more challenging. Therefore, it would be effective to implement actions in person in the classroom to capture their interest.

Moreover, regarding the activities undertaken, girls also provide more negative evaluations than boys, and once again, the worst results are observed in secondary education.

However, the results related to Computational Thinking skills do not show significant differences between the two genders, either in primary or secondary education. What is noticeable is the decline in results after performing activities in the online mode. Hence, future research in this area will focus on integrating these types of activities in in-person settings, with an emphasis on younger girls to prevent them from losing interest in Computer Science as they grow older.

This study shows that the best way to train the future engineers' Computational Thinking skills is in-person, especially in secondary education. One possible reason could be the unavailability of tangible devices in the online model, resulting in a lack of interest and poorer academic performance, as the researcher-teacher is not physically present to guide them in the classroom. This study also shows that there are no significant gender differences in primary, but there are in secondary, so efforts should focus on carrying over this interest from primary education.

Future research will aim to investigate the underlying factors that may cause the online model students to exhibit lower levels of interest compared to the in-person, and explore potential strategies to address this issue.

## ACKNOWLEDGMENTS

The authors would like to acknowledge the "Aula Cultural de Pensamiento Computacional de la Universidad de La Laguna" and the "Unidad de Cultura Científica y de la Innovación (Cienci@ULL) de la Fundación General de la Universidad de La Laguna." This research was supported by the project "Piensa ComputacionULLmente (REF23120270). Educational program to promote computational thinking through activities that enable its development and inclusion in the curriculum," of the Cabildo de Tenerife and the Fundación General de la Universidad de La Laguna, and by the project "Computational thinking in the Canarian educational system: diagnosis and roadmap for its incorporation into the curriculum", with number 2020EDU01 of the Fundación LaCaixa-Cajacanarias. The work of Rafael Herrero-Álvarez has been co-financed by the Canary Islands Agency for Research, Innovation and the Information Society of the Ministry of Universities, Science, Innovation and Culture and by the European Social Fund Plus (ESF+) Integrated Operational Program of the Canary Islands 2021–2027, Axis 3, Priority 74 (85%).

## DATA AVAILABILITY STATEMENT

Research data are not shared.

## ORCID

*Rafael Herrero-Álvarez* http://orcid.org/0000-0003-0650-7155
*Coromoto León* http://orcid.org/0000-0002-9294-9234
*Gara Miranda* https://orcid.org/0000-0002-2181-0067
*Eduardo Segredo* https://orcid.org/0000-0002-4085-4356

## REFERENCES

1. V. Barr, and C. Stephenson, *Bringing computational thinking to K-12: what is involved and what is the role of the computer science education community?* ACM Inroads **2** (2011), no. 1, 48–54. https://doi.org/10.1145/1929887.1929905
2. S. K. Basak, M. Wotto, and P. Bélanger, *E-learning, m-learning and d-learning: conceptual definition and comparative analysis*, E-learn. Digit. Media **15** (2018), no. 4, 191–216. https://doi.org/10.1177/2042753018785180
3. K. Bati, *A systematic literature review regarding computational thinking and programming in early childhood education*, Educ. Inf. Technol. **27** (2021), no. 2, 2059–2082. https://doi.org/10.1007/s10639-021-10700-2
4. M. U. Bers, L. Flannery, E. R. Kazakoff, and A. Sullivan, *Computational thinking and tinkering: exploration of an early childhood robotics curriculum*, Comput. Educ. **72** (2014), 145–157. https://doi.org/10.1016/j.compedu.2013.10.020
5. K. A. Blotnicky, T. Franz-Odendaal, F. French, and P. Joy, *A study of the correlation between stem career knowledge, mathematics self-efficacy, career interests, and career activities on the likelihood of pursuing a stem career among middle school students*, Internat. J. STEM Educ. **5** (2018), no. 1, 22–15. https://doi.org/10.1186/s40594-018-0118-3
6. Z. Bojović, P. D. Bojović, D. Vujošević, and J. Šuh, *Education in times of crisis: rapid transition to distance learning*, Comput. Appl. Eng. Educ. **28** (2020), no. 6, 1467–1489. https://doi.org/10.1002/cae.22318
7. K. Brennan, M. Resnick, and M. M. Lab, *New frameworks for studying and assessing the development of computational thinking*, Proc. 2012 Annu. Meet. American Educational Research Association, AERA, Vancouver, BC, Canada, 2012, p. 25.
8. G. Chen, P. Chen, W. Huang, and J. Zhai, *Continuance intention mechanism of middle school student users on online learning platform based on qualitative comparative analysis method*, Math. Probl. Eng. **2022** (2022), 1–12. https://doi.org/10.1155/2022/3215337
9. Code.org. *Code.org*. https://studio.code.org/courses, 2024.
10. Code.org. *Hour of code*, https://hourofcode.com/es, 2021.
11. Computer Science Education Research Group. *University of Canterbury, Cs unplugged*, https://csunplugged.org, 2021.
12. M. A. Conde, F. J. Rodríguez-Sedano, C. Fernández-Llamas, J. Gonçalves, J. Lima, and F. J. García-Peñalvo, *Fostering steam through challenge-based learning, robotics, and physical devices: a systematic mapping literature review*, Comput. Appl. Eng. Educ. **29** (2021), no. 1, 46–65. https://doi.org/10.1002/cae.22354
13. T. Crick, C. Knight, R. Watermeyer, and J. Goodall, *The impact of Covid-19 and "emergency remote teaching" on the UK computer science education community*, United Kingdom & Ireland Computing Education Research Conference, Glasgow, United Kingdom, 2020, pp. 31–37. https://doi.org/10.1145/3416465.3416472
14. C. Donham, H. A. Barron, J. S. Alkhouri, M. C. Kumarath, W. Alejandro, E. Menke, and P. Kranzfelder, *I will teach you here or there, I will try to teach you anywhere: perceived supports and barriers for emergency remote teaching during the COVID-19 pandemic*, Internat. J. STEM Educ. **9** (2022), no. 1, 19. https://doi.org/10.1186/s40594-022-00335-1
15. J. Fagerlund, P. Häkkinen, M. Vesisenaho, and J. Viiri, *Computational thinking in programming with scratch in primary schools: a systematic review*, Comput. Appl. Eng. Educ. **29** (2021), no. 1, 12–28. https://doi.org/10.1002/cae.22255
16. M. N. Giannakos, L. Jaccheri, and R. Proto, *Teaching computer science to young children through creativity: lessons learned*

from the case of Norway, Proc. 3rd Computer Science Education Research Conference on Computer Science Education Research CSERC'13, Open Universiteit, Heerlen, Netherlands, 2013, pp. 103–111
17. Google, *Teach computer science & coding to kids—cs first*, https://csfirst.withgoogle.com/s/en/home, 2021.
18. A. Heidari, N. J. Navimipour, M. Unal, and S. Toumaj, *The COVID-19 epidemic analysis and diagnosis using deep learning: a systematic literature review and future directions*, Comput. Biol. Med. **141** (2022), 105141. https://doi.org/10.1016/j.compbiomed.2021.105141
19. R. Herrero-Álvarez, C. León, G. Miranda, C. Rodríguez-León, E. Segredo, L. García, and Y. Díaz, *Computational thinking olympiad as an example of service-learning*, 14th Internat. Conf. Education and New Learning Technologies, IATED Academy, Palma, Spain, 2022, pp. 3457–3462.
20. R. Herrero-Álvarez, C. León, G. Miranda, and E. Segredo, *El proyecto piens@ computacion@llmente*, CINAIC 2019, Universidad de Zaragoza, Madrid, Spain, 2019, pp. 573–578.
21. R. Herrero-Álvarez, C. Leon, G. M. Valladares, and E. Segredo, *How do young students get enthusiastic about computational thinking activities?* 13th Internat. Conf. on Education and New Learning Technologies, IATED Academy, 2021, p. 725.
22. R. Herrero-Álvarez, G. Miranda, C. León, and E. Segredo, *Engaging primary and secondary school students in computer science through computational thinking training*, IEEE Trans. Emerg. Top. Comput. **11** (2023), no. 1, 56–69. https://doi.org/10.1109/TETC.2022.3163650
23. K. J. Herrmann, K. Lindvig, and J. Aagaard, *Curating the use of digital media in higher education: a case study*, J. Further Higher Educ. **45** (2021), no. 3, 389–400. https://doi.org/10.1080/0309877X.2020.1770205
24. S. Hiltz, and M. Turoff, *Education goes digital*, Commun. ACM **48** (2005), no. 10, 59–64. https://doi.org/10.1145/1089107.1089139
25. P. Hubwieser, M. Armoni, T. Brinda, V. Dagiene, I. Diethelm, M. N. Giannakos, and M. Knobelsdorf, *Computer science/informatics in secondary education*, Proc. 16th Annu. Conf. Reports on Innovation and Technology in Computer Science Education—Working Group Reports, 2011, pp. 19–38. https://doi.org/10.1145/2078856.2078859
26. R. Israel-Fishelson, A. Hershkovitz, A. Eguíluz, P. Garaizar, and M. Guenaga, *The associations between computational thinking and creativity: the role of personal characteristics*, J. Educ. Comput. Res. **58** (2021), no. 8, 1415–1447. https://doi.org/10.1177/0735633120940954
27. A. Joshi, S. Kale, S. Chandel, and D. Pal, *Likert scale: explored and explained*, Br. J. Appl. Sci. Technol. **7** (2015), no. 4, 396–403. https://doi.org/10.9734/BJAST/2015/14975
28. Joy Labz, *Makey Makey*. https://makeymakey.com/, 2022.
29. Learning Resources, *Code & Go® robot mouse activity set*, https://www.learningresources.com/item-code-gor-robot-mouse-activity-set, 2024.
30. Y. Li, A. H. Schoenfeld, A. A. diSessa, A. C. Graesser, L. C. Benson, L. D. English, and R. A. Duschl, *Computational thinking is more about thinking than computing*, J. STEM Educ. Res. **3** (2020), no. 1, 1–18. https://doi.org/10.1007/s41979-020-00030-2
31. C.-K. Looi, M.-L. How, W. Longkai, P. Seow, and L. Liu, *Analysis of linkages between an unplugged activity and the development of computational thinking*, Comput. Sci. Educ. **28** (2018), no. 3, 255–279. https://doi.org/10.1080/08993408.2018.1533297
32. Makeblock, *mbot*, https://www.makeblock.com/mbot/, 2021.
33. Microsoft, *Microsoft makecode arcade*. https://arcade.makecode.com, 2021.
34. J. del Olmo-Muñoz, R. Cózar-Gutiérrez, and J. A. González-Calero, *Computational thinking through unplugged activities in early years of primary education*, Comput. Educ. **150** (2020), no. 1, 103832. https://doi.org/10.1016/j.compedu.2020.103832
35. I. Panev, *Online teaching in higher education during COVID-19 pandemic*, World J. Educ. Res. **8** (2021), no. 2, 96. https://doi.org/10.22158/wjer.v8n2p96
36. M. Resnick, J. Maloney, A. Monroy-Hernández, N. Rusk, E. Eastmond, K. Brennan, A. Millner, E. Rosenbaum, J. Silver, B. Silverman, and Y. Kafai, *Scratch: programming for all*, Commun. ACM **52** (2009), no. 11, 60–67. https://doi.org/10.1145/1592761.1592779
37. M. Román-González, *Computational thinking test: design guidelines and content validation*, EDULEARN15, IATED, Barcelona, 2015, pp. 2436–2444. https://doi.org/10.13140/RG.2.1.4203.4329
38. M. Román-González, J.-C. Pérez-González, and C. Jiménez-Fernández, *Which cognitive abilities underlie computational thinking? criterion validity of the computational thinking test*, Comput. Hum. Behav. **72** (2017), 678–691. https://doi.org/10.1016/j.chb.2016.08.047
39. M. Román-González, J.-C. Pérez-González, J. Moreno-León, and G. Robles, *Can computational talent be detected? Predictive validity of the computational thinking test*, Internat. J. Child-Comput. Interact. **18** (2018), 47–58. https://doi.org/10.1016/j.ijcci.2018.06.004
40. J.-M. Sáez-López, M. Román-González, and E. Vázquez-Cano, *Visual programming languages integrated across the curriculum in elementary school: a two year case study using "scratch" in five schools*, Comput. Educ. **97** (2016), 129–141. https://doi.org/10.1016/j.compedu.2016.03.003
41. E. Segredo, G. Miranda, and C. León, *Towards the education of the future: computational thinking as a generative learning mechanism*, Educ. Knowl. Soc. **18** (2017), no. 2, 33–58. https://doi.org/10.14201/eks2017182335
42. O. Seneviratne, *Making computer science attractive to high school girls with computational thinking approaches: a case study*, Emerging Research, Practice, and Policy on Computational Thinking, Springer International Publishing, Cham, 2017, pp. 21–32. https://doi.org/10.1007/978-3-319-52691-1_2
43. P. Sengupta, J. S. Kinnebrew, S. Basu, G. Biswas, and D. Clark, *Integrating computational thinking with K-12 science education using agent-based computation: a theoretical framework*, Educ. Inf. Technol. **18** (2013), no. 2, 351–380.
44. R. Smith, and S. Rixner, *Design and evaluation of a collaborative online computational thinking course*, Proc. 2020 ACM Conf. Innovation and Technology in Computer Science Education, ACM, New York, NY, 2020, pp. 342–348. https://doi.org/10.1145/3341525.3387388
45. R. Strachan, A. Peixoto, I. Emembolu, and M. T. Restivo, *Women in engineering: addressing the gender gap, exploring*

*trust and our unconscious bias*, IEEE Global Engineering Education Conference (EDUCON), IEEE, Santa Cruz de Tenerife, Spain, 2018, pp. 2088–2093.

46. Y. M. Tang, P. C. Chen, K. M. Y. Law, C. H. Wu, Y.-p. Lau, J. Guan, D. He, and G. T. S. Ho, *Comparative analysis of student's live online learning readiness during the coronavirus (COVID-19) pandemic in the higher education sector*, Comput. Educ **168** (2021), 104211. https://doi.org/10.1016/j.compedu.2021.104211
47. A. Tucker, F. Deek, J. Jones, D. McCowan, C. Stephenson, and A. Verno, *A model curriculum for K-12 computer science: final report of the ACM K-12 task force curriculum committee*, Technical report, Association for Computing Machinery, Inc (ACM), New York, NY, USA, 2003.
48. A. I. Vodă, C. Cautisanu, C. Grădinaru, C. Tănăsescu, and G. H. S. M. de Moraes, *Exploring digital literacy skills in social sciences and humanities students*, Sustainability. **14** (2022), no. 5, 2483. https://doi.org/10.3390/su14052483
49. M. Webb, T. Bell, Y. J. Katz, N. Reynolds, D. P. Chambers, M. M. Sysło, A. Fluck, M. Cox, C. Angeli, J. Malyn-Smith, J. Voogt, J. Zagami, P. Micheuz, Y. Chtouki, and N. Mori, *Computer science in the school curriculum: issues and challenges*, Tomorrow's Learning: Involving Everyone. Learning with and about Technologies and Computing, AICT-515, Dublin, Ireland, 2018, pp. 421–431. https://doi.org/10.1007/978-3-319-74310-3_43
50. M. Webb, N. Davis, T. Bell, Y. Katz, N. Reynolds, D. Chambers, and M. Sysło, *Computer science in K-12 school curricula of the 2lst century: why, what and when?* Educ. Inf. Technol. **22** (2017), no. 2, 445–468. https://doi.org/10.1007/s10639-016-9493-x
51. J. M. Wing, *Computational thinking*, Commun. ACM **49** (2006), 33–35.
52. E. B. Witherspoon, C. D. Schunn, R. M. Higashi, and E. C. Baehr, *Gender, interest, and prior experience shape opportunities to learn programming in robotics competitions*, Internat. J. STEM Educ. **3** (2016), no. 1, 1. https://doi.org/10.1186/s40594-016-0052-1
53. Z. Xiong, Q. Liu, and X. Huang, *The infiuence of digital educational games on preschool children's creative thinking*, Comput. Educ. **189** (2022), no. 3, 104578. https://doi.org/10.1016/j.compedu.2022.104578

## AUTHOR BIOGRAPHIES

**Rafael Herrero-Álvarez** received a B.S. degree from the Universidad de La Laguna, Spain in 2017, in Computer Science and an M.S. degree from the Universitat Oberta de Catalunya, Spain in 2019, in Information and Communication Technology Security. He is currently working on a PhD in Computer Science related to the Development of an Open-Source Emotion-Based Adaptive Learning Support System using Computational Thinking Activities.

**Coromoto León** received an M.S. degree in Mathematics in 1990 and a Ph.D. degree in Computer Science in 1996, both from the Universidad de La Laguna, Tenerife, Spain, where she is currently a Professor of Computer Languages and Systems. She has led multiple doctoral courses and seminars and has directed several research projects and contracts for innovation and industry transfer. Her research interests include programming languages, algorithmic techniques, optimization, parallel programming, and computational thinking.

**Gara Miranda** received computer science B.S., M.S., and Ph.D. degrees from Universidad de La Laguna (ULL), Spain, in 2002, 2004, and 2009, respectively. She is currently an Associate Professor in the Departamento de Ingeniería Informática y de Sistemas at ULL. The primary focus of her research is tackling combinatorial optimization problems utilizing heuristics, metaheuristics, and hyperheuristics. Additionally, she is also interested in computational thinking and other issues related to computer science teaching and promotion.

**Eduardo Segredo** received his B.S., M.S., and Ph.D. degrees from the Universidad de La Laguna, Spain, in 2006, 2008, and 2014, respectively, all in computer science. He is currently an associate professor in the Departamento de Ingeniería Informática y de Sistemas at Universidad de La Laguna. His current research interests include single/multi/many-objective optimization, evolutionary algorithms, metaheuristics, machine learning, and computational thinking.

**How to cite this article:** R. Herrero-Álvarez, C. León, G. Miranda, and E. Segredo, *Training future engineers: Integrating Computational Thinking and effective learning methodologies into education*, Comput. Appl. Eng. Educ. (2024), e22723. https://doi.org/10.1002/cae.22723