education sciences

MDPI

Article

Introduction to Computational Thinking with Scratch for Teacher Training for Spanish Primary School Teachers in Mathematics

Álvaro Molina-Ayuso [1], Natividad Adamuz-Povedano [2,*], Rafael Bracho-López [2] and Manuel Torralbo-Rodríguez [2]

1 Departamento de Matemáticas, IES Blas Infante, 14007 Córdoba, Spain
2 Departamento de Matemáticas, Universidad de Córdoba, 14071 Córdoba, Spain
* Correspondence: nadamuz@uco.es

Abstract: In recent years, the inclusion of computational thinking in education has become very important. This is a response to the needs of the evolution of our society and the skills demanded in students to obtain practical and integrated training. For this reason, the educational inclusion of these types of practices, strategies, and skills has been the subject of study in recent years. However, it is equally important to prepare and analyse the initial training of future teachers in this area. This research paper presents an empirical experience in which the degree of development of skills associated with computational thinking in preservice primary teachers is examined. For this purpose, programming practices with Scratch were carried out with a total of 149 students of primary education university degrees as part of their training in mathematics education. An experiment was designed for a control group and an experimental group with initial and final measurements using a validated diagnostic instrument consisting of 30 questions associated with computational concepts and their application: a computational thinking test. The result of the experience is positive, as a more significant improvement was observed in the experimental group, which was also accompanied by the impressions, provided by participants, that point in a positive, useful, and practical direction in terms of the development of this type of educational practice being relevant enough to introduce to the teaching and learning process of mathematics.

Keywords: computational thinking; mathematics; primary education; scratch

check for updates

Citation: Molina-Ayuso, Á.; Adamuz-Povedano, N.; Bracho-López, R.; Torralbo-Rodríguez, M. Introduction to Computational Thinking with Scratch for Teacher Training for Spanish Primary School Teachers in Mathematics. *Educ. Sci.* **2022**, *12*, 899. https://doi.org/10.3390/educsci12120899

Academic Editor: Christopher R. Rakes

Received: 6 November 2022
Accepted: 2 December 2022
Published: 8 December 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The importance of developing computational thinking as a curricular element reflects the needs that society is demanding, both from our students and teachers, and has been linked to the technological development that has characterised all sectors of our daily lives [1]. Obviously, not all professions that our students will enter in the near future will be developed using computers or any programmable technology [2], but the current conceptualisation of computational thinking goes beyond learning to program or carrying out processes using a computer: it addresses and develops the processes and skills needed to pose and solve complex problems [3].

When talking about computational thinking, it is essential to refer to the contributions made to the field of computer science by the American engineer Jannette Wing. Her contributions, given in the paper presented at the ACM (Association for Computing Machinery of the United States) in 2006, marked a very important starting point for much of the research carried out over the last 15 years. In this work, Wing establishes computational thinking as a set of universally applicable skills that every student should develop in the educational stage, highlighting the fact that it should not be considered an exclusive competence of a computer scientist [4]. As a background to this important theoretical note on the concept of computational thinking and the importance of incorporating it into the school environment as part of an active curriculum, it is worth highlighting the framework provided by diSessa [5] in his book *Changing Minds*. Here, the author establishes a conceptual basis for approaching computer literacy. He formalises the idea of computers as the basis for a new form of literacy with the power to be introduced across subjects, content, and subject areas. From this point on, many definitions and contextualisations have followed in an attempt to reach a consensus on both the definition and the conceptual framework of computational thinking. In very general terms, Hemmendiger [6] gives a definition for computational thinking as the action of thinking like an economist, a physicist, or an artist and understanding how to use computation to solve problems and create and discover new questions that can be explored. Subsequently, Wing went on to establish a definition by making another more generic contribution, saying that computational thinking is a thinking process used to formulate a problem and express its solution or solutions in terms that a computer can effectively apply [7]. At the end of this decade, the formal definition of computational thinking was still without a definitive consensus. It remained an open topic of ongoing discussion, but there was still agreement that computational thinking includes algorithmic thinking, handling multiple levels of abstraction, decomposing problems into smaller parts, and representing data through models [8]. More recently, making an analysis of research works in the literature on computational thinking and its definition, Moreno et al. [9] proposed a definition in which it is conceived as the ability to formulate and represent problems in order to solve them through the use of the tools, concepts, and practices of the computer science discipline, such as abstraction, decomposition, or the use of simulations.

In reference to a conceptual framework, among others, the one provided by Grover and Pea [10] stands out. In it, computational thinking is positioned as a mediating element in STEM practices. This conceptual framework establishes a series of elements, divided into concepts and practices, which help to understand what computational thinking is and what reasoning process it involves. Computational thinking concepts include logic and logical thinking, algorithms and algorithmic thinking, patterns and pattern recognition, abstraction and generalisation, evaluation, and automation. The computational thinking practices referred to in this conceptual framework attempt to approximate the work that a computer scientist can perform when confronted with the need to problem solve. Included within this category are the decomposition of a problem, the creation of computational artefacts, testing and debugging, repeated improvement, collaboration, and creativity.

In addition to the different successions and conceptions of computational thinking, in order to be able to work effectively in the classroom and carry out an adequate curricular inclusion, it is important to address the assessment of this type of skill in empirical research, using different measurement tools validated by experts that allow teachers to analyse the learning process and the acquisition of a student's skills [11]. If we consider differentiating types of tools for assessing computational thinking, we can find an initial categorisation [12] based on the elements addressed and the approach or format presented for their results, with the following types being differentiated:

1. Summative CP tools, among which aptitude tests and tools for assessing content knowledge can be distinguished.
2. Training-iterative tools are resources that provide automatic feedback to users to improve their computational practices and are usually specific to a programming language or environment.
3. Computational thinking skills transfer tools. These tools look at how computational thinking skills acquired through practice can be applied to different types of real-life problems.
4. Perceptual scales of computational thinking attitudes, which are used to determine the levels acquired in different skills.
5. Computational thinking vocabulary assessments, which are used to measure verbally expressed elements and the dimensions of computational thinking.

Sometimes, it will be interesting to combine different assessment tools in order to have a more complete picture of the acquisition of skills and the learning progress of our students [13].

Regarding the inclusion of computational thinking in educational practices in the process of learning and teaching mathematics, numerous studies have focused on different topics such as geometry and angular measures, fractions, proportions, or problem-solving. For example, Lindh and Holgersson [14] obtained positive results when developing logical and mathematical problem-solving processes using educational robotics kits. Similarly, Nugent et al. [15] showed a positive impact on the learning of mathematics and geospatial concepts in 12–14-year-olds using robotics kits and geolocation technology to develop programming activities.

Similarly, positive results are found in works such as Song [16] when it comes to including practices with Scratch in secondary education to work on aspects related, for example, to trigonometry through a game-based methodology. Calder [17] also showed ways to work on mathematical concepts and ideas in geometry and measurement through Scratch projects. Within the same mathematics topic, geometry, the work of Iskrenovic-Momcilovic [18] analyses the influence that the inclusion of Scratch has on learning basic geometry concepts compared with traditional work on these mathematical contents, making them more interesting and attractive for students.

Chiang [19], for his part, showed positive results in the process of solving equations for secondary education students by working on this subject through an experience based on the construction of a game with Scratch. Taylor et al. [20] showed how the use of Scratch can encourage cooperative work in the problem-solving processes by setting goals and checking ideas worked on in the mathematics learning process. Lie et al. [21] also analysed and showed the relationship established when working on programming concepts with Scratch, mathematical reasoning, and the problem-solving process; Vidal-Silva et al. [22] analysed how the use of Scratch in Primary Education students enables the development of algorithmic reasoning and logical–mathematical skills in an effective way. In the same line, to reinforce the problem-solving process, Molina et al. [23] offered positive evidence of how working with Scratch helps improve the implementation of Polya's method for solving mathematical problems.

These examples show that it is an effective tool for more participatory practices, as it reinforces basic skills in this subject and allows for the development of skills such as the use of patterns, abstraction, decomposition, or algorithmic thinking [24]. But it is also worth highlighting the importance of providing a new scenario for developing these skills; this includes an important motivational factor, which is key for the effective transmission of cognitive skills [25]. This is based on the close link between mathematical reasoning and computational thinking, a relationship that always generates some discussion [26]. This approach makes it possible to develop mathematical content through an integrative and transversal learning experience, making room for the development of competencies beyond the merely mathematical and facilitating much more complete and practical training of our students [27].

The importance of developing computational thinking skills in students at different educational levels is widely accepted, and their academic inclusion is necessary at different levels of both primary and secondary education [11,28,29]. Currently, we can find many studies that analyse different work processes in which these types of skills are developed in a cross-cutting manner and integrated into the curriculum. But for the development of different skills and abilities to be carried out effectively in our students, it is important that teacher training is adequate in order to avoid, among other things, situations where their implementation depends on the personal perspective of each teacher, resulting in multiple practical applications that can have different repercussions in the training of students.

In the report prepared by the European Commission, which analyses the effects of the formal inclusion of computational thinking in different countries of the European Union between 2016 and 2021, it is stated that curricular inclusion should be carried out at all educational levels in a comprehensive manner in order to articulate an integration strategy from primary education to the end of compulsory education in a progressive manner and in accordance with each age group [30]. The importance of teacher training is also raised, especially for those teachers who are less familiar with the use of computational thinking resources, to ensure professional development in this field. Specifically, in the Spanish education system, this aspect is of particular interest in view of the importance of computational thinking in the current educational legislation regulating minimum teaching in primary education [31]. In what can be described as essential skills for any student, which are the key competencies derived from the European reference framework [26], we find digital competence. This competence refers to students acquiring correct problem-solving literacy and computational and critical thinking. In addition, in a more concrete way, this is reflected in the description of different specific competencies by defining the performance of each student in an area, domain, or subject when working with basic knowledge. Within the specifications for the teaching of mathematics, in the new law of education [32], one of the main organisational axes is problem-solving, a process that can be addressed and related to computational thinking [33,34]. As in the general description, it is also included in specific competencies in order to establish a relationship with the output profile. Similarly, this concept of computational thinking appears to define different assessment criteria and basic knowledge in the different cycles of primary education. Therefore, it is understood that there is a need for both initial and continuous training for Primary Education teachers to help them acquire the necessary concepts to develop computational thinking practices in an integrated manner in the curriculum.

Teacher training in this field has certain limitations [35], so it is necessary to update the skills of future teachers in this area [36] in order to adapt to the needs that education demands today, with the overall challenge being the professional development of teachers whose skills enable them to bring computational thinking into the primary and secondary education classroom [37]. There are indications of the benefits of including computational thinking practices in initial teacher training so they can acquire the necessary knowledge in this field and be able to apply it in the classroom. However, this requires the involvement not only of computer science teachers but also of teachers working in different areas who acquire the necessary knowledge in this field [38]. It is important for trainee teachers to analyse and understand what computational thinking is in the context of the subject they want to work on, and it is necessary for them to have a thorough knowledge of both their own discipline and the different concepts of computational thinking that students can learn in the classroom [39]. Therefore, the question of how to develop computational thinking skills in initial teacher education is an important topic for which research is timely and necessary [40].

In this regard, we can find research such as that of Yadav et al. [41], which is based on attitudinal questionnaires to analyse the perception and knowledge that future teachers may have regarding computational thinking, using questions and activities that show the main associated computational concepts. Other studies also analyse these aspects as well as how computational thinking can be integrated into different disciplines [42]. But the present research work, in addition to these types of issues, analyses the degree of development of different computational thinking skills through integrated activities in the context of mathematics teaching in initial teacher education. For this purpose, a series of practices have been carried out with the block programming language Scratch, which allows teachers to work with methodologies based on the use of digital devices for learning mathematics while addressing different computational concepts [24].

To measure the degree of development of these computational thinking skills, a validated tool called the computational thinking test (CT-test) has been used [43]. The aim of this tool is to measure the ability to create and solve problems based on basic programming concepts and the use of the logical syntax of these languages such as loops, functions, variables, or conditionals [44]. This test is one of the most popular tools and is based on an evaluation with a block programming language, although it is not linked to any particular programme [45]. In its basic format, the test address computational concepts such as the use of basic directions and sequences, repeating loops with or without conditionals, simple and compound conditionals (yes, yes–no), "while" conditionals, and simple functions. These concepts are associated with the standards provided by the Computer Science Teacher Association, which are continuously revised and adapted, and based on the Brennan and Resnick [46] framework, cited in the introduction. Depending on the level of education at which the tool is to be used, there are different versions. In the case of higher levels, such as high school or first-year university studies, it is convenient to use the version of the computational thinking test that includes Bebras problems. Such problems are based on the so-called Bebras competition, an international event that takes place online every year and whose aim is to promote computational thinking. It also allows for analysis as to whether computational thinking skills are correctly applied or transferred to everyday situations [47]. The test validation process was carried out with a group of 20 experts, 14 men and 6 women, from various research groups and associations related to the promotion and development of computational thinking in education [29].

If we look at similar studies carried out in Spain, we can highlight the one by Esteve-Mon et al. [36], whose work showed positive results in the development of computational thinking after carrying out an intervention with educational robotics resources with unplugged, playing, making, and remixing activities with university education students. In a similar vein, Dodero et al. [48] discussed the impact of the inclusion of block programming resources in the initial training of teachers of different disciplines, highlighting the potential that future teachers see in this type of tool. In this sense, González and Muñoz-Repiso [49] highlighted the motivation of future teachers to include educational robotics resources in early childhood education to achieve curricular objectives. In order to continue in the same line of work as previous experiences, this paper analyses the degree of acquisition of the competencies associated with computational thinking by students in the second year of their primary education degrees as part of the didactics of numerical operations and measurement subject. The main aim of this study is to see to what extent the necessary knowledge of this subject is acquired through various programming practices using the visual programming language Scratch and the conception that future teachers have of the application of this type of practice with primary education students. Likewise, as part of initial teacher training, the aim of this type of practice is to promote new ways of teaching mathematics in the classroom through educational practices with a transversal and integrated approach.

2. Materials and Methods

The starting point of this work is to analyse the degree of development of computational thinking skills in primary education students working with Scratch. It also analyses the perspective that the use of this type of resource and the development of the associated skills benefits future teachers when looking for new ways of teaching mathematics in accordance with the needs of today's society. To analyse these aspects, an empirical experience was designed in which the participants had to carry out a series of practices with Scratch and some error correction challenges.

The specific objectives of this work are as follows:

- O1. To analyse the degree of development of different computational competencies by carrying out a series of practices with Scratch.
- O2. To assess the perception of these types of educational practices and resources in the initial training of future teachers.
- O3. To promote the use of new methodologies and learning strategies by working on computational thinking in a transversal way in the teaching of mathematics.

To carry out this empirical process, we worked with a mixed methodology [50] in which the quantitative analysis was carried out with the CT-test. The aim of this tool is to measure the ability to create and solve problems based on basic programming concepts and the use of the logical syntax of these languages such as loops, functions, variables, or conditionals [44]. As we mentioned before, this test is one of the most popular tools based on an evaluation with a block programming language, although it is not linked to any particular programme [45]. The test is composed of 28 multiple-choice questions differentiated into categories that address computational concepts such as the use of basic directions and sequences, repeating loops with or without conditionals, simple and compound conditionals (yes, yes–no), "while" conditionals, and simple functions. The test is an individual response test in which each question has four possible options with only one correct one. For this experiment, we used the Bebras composite CT-test with Bebras tasks, consisting of a total of 30 questions. To create this test from the original CT-test, we selected the 20 most discriminative items [51] and added 4 extra difficulty items and a selection of 6 Bebras tasks [47]. These tasks are not a measuring instrument per se; they do not measure aptitude, but they are valid for measuring "competence" in a sense similar to the way different skills are measured in a PISA test [52]. According to Román-González [51], these tasks do not focus on measuring computational thinking in a pure aptitudinal state, but rather on measuring how the subject is able to transfer this computational thinking to everyday and meaningful problems.

These data are supported by those obtained from a qualitative process using a survey to find out the impressions of the participants, which will help to know the classroom reality from the point of view of its protagonists, both students and teachers, in order to understand our object of study by analysing the perceptions and interpretations of the people involved in this work [53]. This combination of qualitative and quantitative techniques provides results characterised by a complementary perspective for the analysis of the different objects of study characterised in this work [54].

The empirical process was developed using a quasi-experimental design, with a non-equivalent control group and pre-test and post-test measures [55]. The qualitative process was carried out with a questionnaire in which aspects referring to the development of the practices, the characteristics of each of the tasks carried out, and general aspects of the use and application of computational thinking practices for primary education were collected.

The study was carried out with a total of 149 students of the primary education degree programme taking the didactics of numerical operations and measurement subjects. The sample is distributed as shown in Table 1:

Table 1. Distribution of participants.

	Men	**Women**	**Total**
Control group	17	53	70
Experimental group	20	59	79
Total	37	112	149

Participants have a basic level of technological knowledge, in general aspects, but no specific knowledge of computational thinking or related practices in this field.

Development of the Experience

For the experimental group, the experience was carried out for a total of five sessions: two of 90 min and three of one hour each. In the first session, an introduction was given to explain what computational thinking is and its importance as an educational resource for primary education. Subsequently, a pre-test followed using the tool mentioned above, and during the following three sessions, practical work with Scratch was carried out. Firstly, there was an activity called Animate Your Name, the aim of which was simply to familiarise students with the working environment while introducing simple programming elements such as repetition loops. With a mathematical approach, elements such as angular measurements or the coordinates of the Cartesian plane were also used to define certain animations. In the second practice, participants learned to define interactive elements and to work with concepts such as parallelism or conditional "if, if not" loops. The third practice consisted of creating an animated story but focused on drawing a square on the screen from the coordinates of the four vertices. To do this, the participants used different functions of the programme while working on a very important concept in the field of programming and problem-solving: decomposition. Finally, they practised creating a game to hunt for objects that appear randomly on the screen. This simple idea allowed them to work with the use of variables to store the score obtained, parallelism, conditional loops, and plane coordinates, among other things. All the practices were focused on addressing the different concepts that are worked on in the computational thinking test in an integrated way and with activities that can be contextualised in the educational field since one of the objectives of the empirical experience is to promote the use of new methodologies and learning strategies in mathematics by developing computational thinking.

From the second practical session onwards, the students had to solve a series of exercises designed to correct errors in a code similar to the activities proposed in class. The aim was to find a more autonomous application for what they had learnt in the practical sessions so that they could internalise the concepts they had worked on.

The final session was used to raise doubts, correct mistakes made by some participants, and complete the final test.

The control groups took the initial and final test on the same dates as the experimental groups thanks to coordination between the teachers of both groups.

3. Results

The results obtained by both the experimental and control groups in the test carried out at the beginning and end of the experiment can be seen in Figures 1 and 2. The boxplot data (Figure 2) are presented in Table 2.

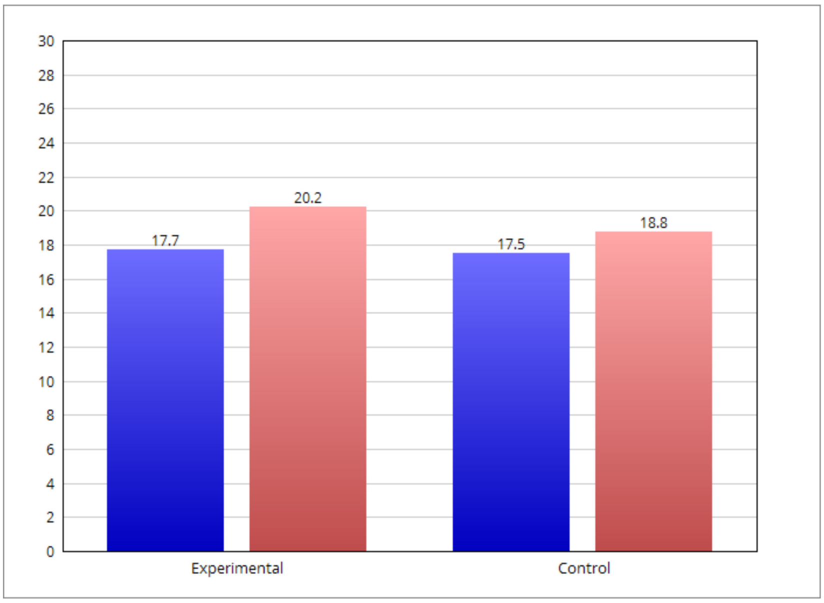

Figure 1. Average scores obtained by participants with the CT-test (score range: 0–30), initial test on blue, final test on red.

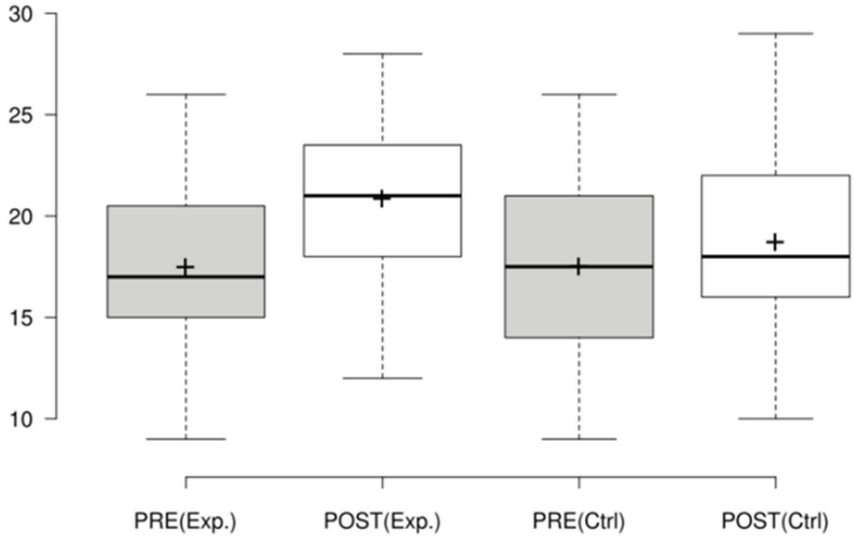

Figure 2. Boxplot of the scores (see Table 2) obtained.

Table 2. Boxplot data of CT-test scores.

	PRE (Exp.)	POST (Exp.)	PRE (Ctrl.)	POST (Ctrl.)
Upper End	26	28	26	29
Lower End	9	12	9	10
1st Quartile	15	18	14	16
3rd Quartile	20.5	23.5	21	22
Medium	17	21	17.5	18

The results show that in both the experimental and control groups there was an improvement in the score obtained in the final test compared with the initial one, although that of the experimental group was more significant. It is understood that the group used as the control group will always show some improvement simply by taking the test a second time, as the participants have the chance to ask themselves again about some doubts they may have had when taking the test and improve some of their answers. The average difference between the initial and final test for the control group was 1.3, while for the experimental group, it was 2.5, so the practical intervention clearly had a significant effect on the development of the skills analysed with this test. The data of the experimental group obtained in the pre-test and post-test have a standard deviation of 3.7 and 3.9 points, respectively, while the data of the control group obtained in both tests have a standard deviation of 4.2 and 4.4 points each.

The computational thinking test has questions differentiated by the use of directions, loops, conditionals, and functions, which allows for a more detailed observation of the results. With this, we saw that, in the final test, a greater number of correct answers were been given by the experimental group, and in comparison with the control group, in the questions referring to directions and loops. These correspond to the first 12 questions of the test, and it can be understood that these are the structures and instructions that have been repeated the most in the practices carried out during the practical phase.

3.1. Statistical Assumptions

In order to consider the validity of these data, the values obtained in the corresponding statistical analysis are shown below. Using PSPP statistical software, the difference between the data obtained in the initial and final test was analysed to obtain information regarding the homogeneity of variances and their normal distribution (Table 3). Related to homogeneity, we obtained a Levene's statistic of 3.29 with a significance value of 0.072, which is greater than 0.05 (p-value of statistical significance) [56]. With these data, we accepted that the distribution of the control and experimental groups adjust to a normal distribution and that there is a similarity between the variances given the significance value obtained in Levene's test.

Table 3. Normal distribution of groups.

	Value
N	149
Media	2.39
Standard deviation	3.39
More extreme differences absolute	0.12
More extreme differences positive	0.12
More extreme differences negative	−0.08
Kolmogorov–Smirnov's Z	1.42
Sig. asst. (two-tailed)	0.026

Likewise, a normal distribution of the differences obtained between the two groups can be accepted thanks to the value obtained in the Kolmogorov–Smirnov test, with an asymptotic significance of 0.026, which is less than 0.05. Therefore, a normal distribution of this sample can be accepted, having obtained a confidence interval of 95%.

In addition, the independence between the observations of each group can be accepted, as well as the random observation within each group, since we worked with separate groups of students who did not have any interactions in the usual daily work in the classroom or during the development of the empirical experience. Since the homogeneity of the groups, the normality of the samples, and the independence of the observations have been established, to conclude, these differences are significant, and a comparison should be made with the one-factor analysis of variance (ANOVA) [57]. When comparing the groups, a significance value of less than 0.05 was obtained (Table 4), which indicates that the difference between the two groups is statistically significant, since a 95% confidence interval rules out the null hypothesis that the groups are equal.

Table 4. ANOVA analysis.

	Sum of Squares	**Df**	**Mean Square**	**F**	**Sign.**
Between groups	176.34	1	176.34	16.97	0.000
Intra groups	1527.81	147	10.39		
Total	1704.15	148			

3.2. Qualitative Evidence

As mentioned above, in the work process, the quantitative data and the results obtained are supported by qualitative evidence obtained through a questionnaire in which the participants were able to provide their impressions on different aspects related to the development of the experience. This combination is the pillar of a mixed research method [58] and helps researchers understand and analyse a situation from a multidimensional perspective with the necessary reliability and validity [59]. Therefore, thanks to the responses collected from the participants, we can provide consistency to the positive results obtained in the quantitative process. The vast majority of the questions reflect a very good reception of the practices carried out by the participants, as they see it as a very interesting resource to be included in a transversal way in the last years of primary education. Below, we highlight some of the comments collected:

> "It will enable children to develop critical thinking and problem-solving strategies that they may encounter throughout their school years and life" (Student 27)

> "It is also enhanced by increased creativity, communication, participation and motivation" (Student 39)

In addition, this vision can be extended to include it as a resource for working on mathematics content, with the following being the most noteworthy of the responses collected:

> "With it we can create games to learn mathematics, and thus motivate students in this subject, in addition, with the basic Scratch controls such as moving towards x number, negative and positive numbers for commands, among others, we are already working on mathematics" (Student 22)

It is understood that future teachers see the potential of this resource to help develop computational thinking skills in the process of teaching and learning mathematics learning from an integrated perspective. An important aspect that has helped us understand the development of the practices carried out was whether it was easy or difficult to work with this tool. The participants offered different opinions, such as the following:

> "At the beginning it was difficult for me to use the application because I didn't know it and I had never used it before, but once you start creating projects, you get to know how it works a bit more and it is not so complicated" (Student 6)

This suggests that they see this type of resource as important in their teacher training.

> "I consider it to be fundamental, as it provides us with a series of knowledge and technological resources suitable for the creation of a series of situations that allow us to connect with students and provide them with an education adapted to change" (Student 43)

Finally, it is important to know whether learning and developing these kinds of skills during initial teacher education will have an impact on their future work. The vast majority of the participants said that, yes, they want to use them in the higher grades of primary education when they are teaching.

> "Definitely. In fact, in my opinion, it is a resource that can be used in all grades, even with the supervision of a teacher or a close relative, because of its high percentage of enriching learning" (Student 8)

> "Yes, because they will be able to be closer to ICTs, which are so present in their daily lives, and address mathematical questions through them" (Student 28)

4. Discussion and Conclusions

With all that has been explained in the previous sections, the positive effect of the inclusion of practices with Scratch to promote computational thinking skills in primary school students is clear. In general, based on the results of the computational thinking test, general computational thinking skills applied to the use of a visual or block programming language have been improved. Thanks to the activities carried out in the practical phase, the positive differences between the initial and final test regarding the use of instructions to use or define directional movements and the use of repetition structures or loops stand out.

In addition to this result, which refers to the first objective of this empirical study, the perception that future teachers have of this type of educational resource and the importance of its transversal inclusion for learning mathematics is also in line with this. Seeing that they positively developed the competencies needed to implement these educational strategies and identified them as an important factor for a complete and integrated education in the social context for our pupils, this suggests that this type of content is important for initial teacher training.

Coinciding with other studies that indicate that the technological practices for trainee teachers are beneficial for subsequent implementation in the school environment [60], these results point to a need to include the development of computational thinking and to learn about educational resources that facilitate it within the curricula of future teachers in primary education. This approach is related to the words of McComas [61], who points out that it is an imperative need in our society to prepare teachers to effectively develop competencies and address content through an appropriate curriculum. This idea is complemented by the conclusions of Gabriele et al. [62], who extend this to initial teacher training in order to prepare them to develop 21st-century competencies and skills in our students. Furthermore, participants have highlighted the beneficial aspect of this type of practice, as they favour the development and implementation of innovative learning methodologies and strategies, with teacher training being a very important context for this point [63,64].

For future research, it would be interesting to replicate this type of intervention with primary education students but broaden the methodological approach to analyse how they can carry out the implementation of this type of resource in the classroom during a teaching practice period. This could help give more solidity to the conclusions obtained regarding the skills developed in initial teacher training on computational thinking and the best way to apply it in the classroom. Moreover, this establishes a starting point for linking initial teacher training with the development of their work in the classroom and the need for adequate continuous training by different educational administrations.

Author Contributions: Conceptualization, Á.M.-A., N.A.-P., R.B.-L. and M.T.-R.; Methodology, Á.M.-A., N.A.-P. and R.B.-L.; Formal analysis, Á.M.-A.; Investigation, Á.M.-A. and N.A.-P.; Writing—original draft, Á.M.-A.; Writing—review & editing, N.A.-P., R.B.-L. and M.T.-R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki, and approved by the Ethics Committee on Human Research (ECHR) from the University of Córdoba (BOUCO 2019/00189).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Del Venegas, J.J.; Díaz, L.A.; Tosina, R.Y.; Ramos, V.L. La Dimensión Educativa De La Robótica: Del Desarrollo Del Pensamiento Al Pensamiento Computacional En El Aula. *Campo Abierto* **2021**, *40*, 221–233. [CrossRef]
2. Segura, J.A.; Nebot, M.Á.L.; Mon, F.E.; Novella, M.G.V. El debate sobre el pensamiento computacional en educación. *Rev. Iberoam. Educ. Distancia* **2019**, *22*, 171. [CrossRef]
3. Wing, J.M. Computational thinking and thinking about computing. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* **2008**, *366*, 3717–3725. [CrossRef] [PubMed]
4. Wing, J.M. Computational Thinking. *Commun. ACM* **2006**, *49*, 33–35. [CrossRef]
5. Di Sessa, A. *Changing Minds: Computers, Learning, and Literacy*; MIT Press: Cambridge, MA, USA, 2000.
6. Hemmendinger, D. A plea for modesty. *ACM Inroads* **2010**, *1*, 4–7. [CrossRef]
7. Wing, J. *Computational Thinking Benefits Society*; Social Issues in Computing 40th Anniversary Blog; Academic Press: New York, NY, USA, 2014; Available online: http://socialissues.cs.toronto.edu/index.html%3Fp=279.html (accessed on 16 March 2021).
8. Jacob, S.R.; Warschauer, M. Computational Thinking and Literacy. *J. Comput. Sci. Integr.* **2018**, *1*, 1. [CrossRef]
9. Moreno-León, J.; Robles, G.; Román, M.; Rodríguez, J.D. No es lo mismo: Un análisis de red de texto sobre definiciones de pensamiento computacional para estudiar su relación con la programación informática. *Rev. Interuniv. Investig. Tecnol. Educ.* **2019**, *7*, 26–35. [CrossRef]
10. Grover, S.; Pea, R. Computational thinking: A competency whose time has come. *Comput. Sci. Educ. Perspect. Teach. Learn. Sch.* **2018**, *19*, 20–35.
11. Grover, S.; Pea, R. Computational Thinking in K-12: A Review of the State of the Field. *Educ. Res.* **2013**, *42*, 38–43. [CrossRef]
12. Román-González, M.; Moreno-León, J.; Robles, G. Complementary Tools for Computational Thinking Assessment. *Proc. Int. Conf. Comput. Think. Educ.* **2017**, 154–159.
13. Grover, S. "Systems of Assessments" for Deeper Learning of Computational Thinking in K-12. In Proceedings of the Annual Meeting of the American Educational Research Association, Chicago, IL, USA, 20 April 2015.
14. Lindh, J.; Holgersson, T. Does lego training stimulate pupils' ability to solve logical problems? *Comput. Educ.* **2007**, *49*, 1097–1111. [CrossRef]

15. Nugent, G.; Barker, B.; Grandgenett, N.; Adamchuk, V. The use of digital manipulatives in K-12: Robotics, GPS/GIS and programming. In Proceedings of the 39th IEEE Frontiers in Education Conference, San Antonio, TX, USA, 18–21 October 2009; pp. 1–6. [CrossRef]
16. Song, J. Effects of Learning through Scratch-Based Game Programming on Students' Interest in and Perceived Value of Mathematics Curriculum. *J. Korean Assoc. Inf. Educ.* **2017**, *21*, 199–208. [CrossRef]
17. Calder, N. Using Scratch: An integrated problem-solving approach to mathematical thinking. *Aust. Prim. Math. Classr.* **2010**, *15*, 9–14.
18. Iskrenovic-Momcilovic, O. Improving Geometry Teaching with Scratch. *Int. Electron. J. Math. Educ.* **2020**, *15*, em0582. [CrossRef] [PubMed]
19. Chiang, F.-K.; Qin, L. A Pilot study to assess the impacts of game-based construction learning, using scratch, on students' multi-step equation-solving performance. *Interact. Learn. Environ.* **2018**, *26*, 803–814. [CrossRef]
20. Taylor, M.; Harlow, A.; Forret, M. Using a Computer Programming Environment and an Interactive Whiteboard to Investigate Some Mathematical Thinking. *Procedia Soc. Behav. Sci.* **2010**, *8*, 561–570. [CrossRef]
21. Lie, J.; Hauge, I.; Meaney, T. Computer Programming in the Lower Secondary Classroom: Mathematics Learning. *Ital. J. Educ. Technol.* **2017**, *25*, 27–35. [CrossRef]
22. Vidal-Silva, C.; Jiménez, C.; Yupanqui, M.T.; Sepúlveda, S.; Madariaga, E. Development of algorithmic and mathematical-logic competences of children in Chile with scratch. *Int. J. Sci. Technol. Res.* **2020**, *9*, 381–385.
23. Molina-Ayuso, Á.; Adamuz-Povedano, N.; Bracho-López, R. La resolución de problemas basada en el método de Polya usando el pensamiento computacional y Scratch con estudiantes de Educación Secundaria. *Aula Abierta* **2020**, *49*, 83–90. [CrossRef]
24. Resnick, M. Sowing the Seeds for a more Creative Society. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Boston, MA, USA, 4–9 December 2009; p. 30. [CrossRef]
25. Jaeggi, S.M.; Buschkuehl, M.; Shah, P.; Jonides, J. The role of individual differences in cognitive training and transfer. *Mem. Cognit.* **2013**, *42*, 464–480. [CrossRef]
26. Kotsopoulos, D.; Floyd, L.; Nelson, V.; Makosz, S. Mathematical or Computational Thinking? An Early Years Perspective. In *Mathematical Learning and Cognition in Early Childhood*; Springer International Publishing: Cham, Switzerland, 2019; pp. 79–90.
27. Rodríguez-Martínez, J.A.; González-Calero, J.A.; Sáez-López, J.M. Computational thinking and mathematics using Scratch: An experiment with sixth-grade students. *Interact. Learn. Environ.* **2019**, *28*, 316–327. [CrossRef]
28. Barr, B.V.; Stephenson, C. Bringing Computational Thinking to K-12: What is Involved and What is the Role of the Computer Science Education Community? *ACM Inroads* **2011**, *2*, 48–54. [CrossRef]
29. Yadav, A.; Hong, H.; Stephenson, C. Computational Thinking for All: Pedagogical Approaches to Embedding 21st Century Problem Solving in K-12 Classrooms. *TechTrends* **2016**, *60*, 565–568. [CrossRef]
30. Stefania, B.; Augusto, C.; Panagiotis, K.; Valentina, D.; Patricia, W.; Katja, E.; Jeffrey, E.; Anna, H.M.; Eglė, J.; Chiara, M.; et al. *Reviewing Computational Thinking in Compulsory Education. State of Play and Practices from Computing Education*; Publications Office of the European Union: Luxembourg, 2022.
31. Ministerio de Educación y Formación Profesional. Real Decreto 157/2022, de 1 de Marzo, por el que se Establecen la Ordenación y las Enseñanzas Mínimas de la Educación Primaria. pp. 1–109. 2022. Available online: https://www.boe.es/buscar/act.php?id=BOE-A-2022-3296 (accessed on 15 March 2022).
32. Rupérez, F.L. El enfoque del currículo por competencias. Un análisis de la LOMLOE. *Rev. Española Pedagog.* **2022**, *80*, 55–68. [CrossRef]
33. Cui, Z.; Ng, O.-L. The Interplay between Mathematical and Computational Thinking in Primary School Students' Mathematical Problem-Solving within a Programming Environment. *J. Educ. Comput. Res.* **2021**, *59*, 988–1012. [CrossRef]
34. Ng, O.-L.; Cui, Z. Examining primary students' mathematical problem-solving in a programming context: Towards computationally enhanced mathematics education. *ZDM Math. Educ.* **2021**, *53*, 847–860. [CrossRef]
35. Demartini, C.G.; Benussi, L.; Gatteschi, V.; Renga, F. Education and Digital Transformation: The "Riconnessioni" Project. *IEEE Access* **2020**, *8*, 186233–186256. [CrossRef]
36. Esteve-Mon, F.M.; Llopis, M.Á.; Adell-Segura, J. Digital Competence and Computational Thinking of Student Teachers. *Int. J. Emerg. Technol. Learn.* **2020**, *15*, 29. [CrossRef]
37. Espinosa, M.P.P.; Cartagena, F.C. Tecnologías avanzadas para afrontar el reto de la innovación educativa. *RIED Rev. Iberoam. Educ. Distancia* **2020**, *24*, 35. [CrossRef]
38. Yadav, A.; Mayfield, C.; Zhou, N.; Hambrusch, S.; Korb, J.T. Computational Thinking in Elementary and Secondary Teacher Education. *ACM Trans. Comput. Educ.* **2014**, *14*, 1–16. [CrossRef]
39. Yadav, A.; Gretter, S.; Good, J.; Mclean, T. Emerging Research, Practice, and Policy on Computational Thinking. In *Emerging Research, Practice, and Policy on Computational Thinking*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 205–220.
40. Gretter, S.; Yadav, A. Computational Thinking and Media & Information Literacy: An Integrated Approach to Teaching Twenty-First Century Skills. *TechTrends* **2016**, *60*, 510–516. [CrossRef]
41. Yadav, A.; Zhou, N.; Mayfield, C.; Hambrusch, S.; Korb, J.T. Introducing computational thinking in education courses. In Proceedings of the 42nd ACM Technical Symposium on Computer Science Education, Dallas, TX, USA, 9–12 March 2011; pp. 465–470. [CrossRef]

42. Ateşkan, A.; Hart, D.O. Demystifying computational thinking for teacher candidates: A case study on Turkish secondary school pre-service teachers. *Educ. Inf. Technol.* **2021**, *26*, 6383–6399. [CrossRef]
43. Román-González, M. Computational Thinking Test: Design Guidelines and Content Validation. In Proceedings of the EDULEARN15, Barcelona, Italy, 6–8 July 2015; pp. 2436–2444. [CrossRef]
44. Chan, S.W.; Looi, C.K.; Sumintono, B. Assessing computational thinking abilities among Singapore secondary students: A Rasch model measurement analysis. *J. Comput. Educ.* **2020**, *8*, 213–236. [CrossRef]
45. Cutumisu, M.; Adams, C.; Lu, C. A Scoping Review of Empirical Research on Recent Computational Thinking Assessments. *J. Sci. Educ. Technol.* **2019**, *28*, 651–676. [CrossRef]
46. Brennan, K.; Resnick, M. Using Artifact-Based Interviews to Study the Development of Computational Thinking in Interactive Media Design. In Proceedings of the Annual American Educational Research Association Meeting, Vancouver, BC, Canada, 13–17 April 2012.
47. Dagiene, V.; Futschek, G. Bebras international contest on informatics and computer literacy: Criteria for good tasks. *Lect. Notes Comput. Sci.* **2008**, *5090*, 19–30. [CrossRef]
48. Dodero, J.M.; Mota, J.M.; Ruiz-Rube, I. Bringing computational thinking to teachers' training. In Proceedings of the 5th International Conference on Technological Ecosystems for Enhancing Multiculturality, Cádiz, Spain, 18–20 October 2017; pp. 1–6. [CrossRef]
49. González, Y.A.C.; Muñoz-Repiso, A.G.-V. Development of computational thinking and collaborative learning in kindergarten using programmable educational robots. In Proceedings of the 5th International Conference on Technological Ecosystems for Enhancing Multiculturality, Cádiz, Spain, 18–20 October 2017; pp. 1–6. [CrossRef]
50. Johnson, B.; Onwuegbuzie, A.J.; Turner, L.A. Toward a Definition of Mixed Methods Research. *J. Mix. Methods Res.* **2007**, *1*, 112–133. [CrossRef]
51. Román-González, M. Codigoalfabetización y Pensamiento Computacional en Educación Primaria y Secundaria: Validación de un Instrumento y Evaluación de Programas. Ph.D. Thesis, Universidad Nacional de Educación a Distancia, Madrid, Spain, 2016.
52. Hubwieser, P.; Mühling, A. Playing PISA with Bebras. In Proceedings of the 9th Workshop in Primary and Secondary Computing Education on - WiPSCE '14, New York, NY, USA, 5 November 2014; pp. 128–129. [CrossRef]
53. Colás, P.; Buendía, L. *Investigación Educativa*; Alfar: Sevilla, Spain, 1992.
54. Morse, L.; Niehaus, J.M. *Mixed Method Design: Principles and Procedures*; Left Coast Press Inc.: Walnut Creek, CA, USA, 2009.
55. Campbell, D.T.; Stanley, J.C. *Experimental and Quasi-Experimental Designs for Research*; Houghton Mifflin Company: Chicago, IL, USA, 1966.
56. Montgomery, D.C. *Design and Analysis of Experiments*; Wiley: New York, NY, USA, 1991.
57. Castañeda, M.; Cabrera, A.; Navarro, Y.; Vries, W. *Procesamiento de Datos y Análisis Estadísticos Utilizando SPSS*; Edipucrs: Porto Alegre, Brazil, 2010.
58. Pericás, J.V.; Colina, C.L. *Introducción a la Investigación Cualitativa*; Sintesis: Madrid, Spain, 2014.
59. Arias, M.d.M.; López, M.Á.C.; López, J.C.C. *Métodos de Investigación en Psicología*; EOS Universitaria: Madrid, Spain, 2014.
60. Vaca-Cardenas, L.; Tavernise, A.; Bertacchini, F.; Gabriele, L.; Valenti, A.; Pantano, P.; Bilotta, E. An Educational Coding Laboratory for Elementary Pre-service Teachers: A Qualitative Approach. *Int. J. Eng. Pedagog.* **2016**, *6*, 11–17. [CrossRef]
61. McComas, W. *The Language of Science Education: An Expanded Glossary of Key Terms and Concepts in Science Teaching and Learning*; Sense Publishers: Rotterdam, The Netherlands, 2013.
62. Gabriele, L.; Bertacchini, F.; Tavernise, A.; Vaca-Cárdenas, L.; Pantano, P.; Bilotta, E. Lesson Planning by Computational Thinking Skills in Italian Pre-service Teachers. *Informatics Educ.* **2019**, *18*, 69–104. [CrossRef]
63. Duo-Terron, P. STEAM in Primary Education. Impact on Linguistic and Mathematical Competences in a Disadvantaged Context. *Front. Educ.* **2022**, *7*, 1–14. [CrossRef]
64. Moreno-Morilla, C.; Guzmán-Simón, F.; García-Jiménez, E. Digital and information literacy inside and outside Spanish primary education schools. *Learn. Cult. Soc. Interact.* **2020**, *28*, 100455. [CrossRef]