education sciences

MDPI

*Article*

# The Impact of Computational Thinking on Logical-Mathematical Reasoning in High School Education: A Quasi-Experimental Study

**Rubén Pajares Pescador [1,2,*], Luis Jorge Martín-Antón [2,3] and Miguel Á. Carbonero-Martín [2,3]**

[1] Liceo Español Luis Buñuel, 92200 Neuilly-sur-Seine, France
[2] Faculty of Education and Social Work, University of Valladolid, 47011 Valladolid, Spain; luisjorge.martin@uva.es (L.J.M.-A.); miguelangel.carbonero@uva.es (M.Á.C.-M.)
[3] Excellence Research Group GR179 Educational Psychology, Department of Psychology, University of Valladolid, 47011 Valladolid, Spain
* Correspondence: ruben.pajares@educacion.gob.es

**Abstract**

This article explores what impact a program of IT programming (Scratch and Python) activities has on the logical-mathematical reasoning of high school children. The sample comprised a group of 388 high school students, spanning first-year compulsory education (12 years old) up to second-year upper secondary (17 years old) students who were given the Differential and General Skills Battery (BADyG) before and after intervention through a quasi-experimental design. Statistical analysis of data involved multivariate analysis of variance (MANOVA), taking education level and group as independent variables. Results showed a principal significant effect of the group, with greater gains amongst the experimental group in four factors of BADyG (logical reasoning, numerical reasoning, visual-spatial reasoning, and general intelligence). Significant differences were also found in terms of education level in the numerical, visual-spatial, and general intelligence factors, with higher gains observed in the early years of compulsory secondary education compared to upper secondary education. No significant effects were found in the interaction between level and group, such that the impact of intervention was consistent throughout all the school years. In sum, the results suggest that a program which promotes computational thinking can favor the development of certain skills related to logical-mathematical reasoning in high school education under the conditions of this study.

**Keywords:** computational thinking; logical-mathematical reasoning; IT programming; Scratch; Python; high school education

Check for updates

Academic Editor: Jason Morphew

Received: 19 December 2025
Revised: 30 January 2026
Accepted: 17 February 2026
Published: 21 February 2026

**Copyright:** © 2026 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.

## 1. Introduction

### *1.1. Definition of Computational Thinking*

In the 1980s, Papert (1980, 1993) introduced new approaches through his LOGO language that were designed to enhance thinking by using programming. Wing (2008) later took up these proposals and suggested that computational thinking should form part of the school curriculum. Since then, studies exploring this skill have become increasingly common and agree that it is a key competence for all students in the 21st century (Weintrop et al., 2016). Although several different theoretical frameworks have been put forward, the conceptualization of the term remains the subject of debate. Nevertheless, one leading proposal is that of Brennan and Resnick (2012), who posit a three-dimensional

model: (a) computational concepts (sequences, loops, conditionals, etc.); (b) computational practices (iteration, debugging, and abstraction); and (c) computational perspectives (expressing, connecting, and questioning).

Lye and Koh (2014) conceive computational thinking as a competence that encompasses programming skills, designing solutions, process analysis, and documenting. Weintrop et al. (2016) proposed a taxonomy that defines the concept considering four types of practices: data analysis, modeling and simulation, computational problem solving, and systemic thinking.

Barr and Stephenson (2011) consider that, beyond what is strictly the field of IT, computational thinking should be a cross-cutting skill for problem solving. Viewed from this perspective, it is important to analyze, organize, and represent data in a logical manner, to automate solutions with algorithms, use abstractions and models, communicate processes and results, recognize patterns, generalize, and transfer them in different learning contexts. The International Society for Technology in Education (ISTE) and the Computer Science Teacher Association (CSTA) define this concept as a problem-solving process that includes the logical organization of information and data analysis, representation of information through abstraction with models and simulations, solution automation, and transfer of processes to real-life situations (ISTE & CSTA, 2011). They also highlight that such skills help to boost attitudes such as confidence, perseverance, and tolerance towards ambiguity, which are key aspects when solving mathematical problems and in autonomous learning.

The definition proposed by Selby (2015) includes concepts such as abstraction, decomposition, algorithmic thinking, assessment, and generalization. Finally, Shute et al. (2017) underline that computational thinking is a skill that is applicable to many areas, and they define it as a combination of cognitive skills such as decomposition of problems, abstraction, algorithmic thinking, logical reasoning, assessment, and debugging. From an educational standpoint, computational thinking is being incorporated into school curricula in the areas of STEM due to interest in promoting logical-mathematical reasoning (Hsu et al., 2018; Kalelioğlu, 2015).

### *1.2. Skills, Attitudes, and Benefits of Computational Thinking*

Although there is still no universally accepted definition of computational thinking, there is wide agreement concerning its link to an array of skills, attitudes, and benefits (Kallia et al., 2021). Grover and Pea (2013) claim that becoming proficient at programming can boost independent learning and foster a proactive attitude amongst students. According to Wing (2008), computational thinking promotes analytical skills when solving problems and can prove equally important to being taught how to read and write. The author also states that it helps with identifying patterns, abstraction, and the re-use of efficient procedures over a range of different contexts, without the need to reformulate a solution from scratch in each situation.

Computing's formal fundamentals lie in mathematics, such that computational thinking involves the use of logical-mathematical reasoning, since logical connectors, arithmetical operations, and structured problem solving are used. Barr and Stephenson (2011) state that this type of thinking requires a high degree of abstraction, which favors a more abstract, analytical, and efficient way of thinking. The skills highlighted by the authors include decomposition, the use of algorithms to correctly solve problems, debugging possible faults, generalization to identify patterns that are applicable to other problems and, finally, automatization. All these skills are not only crucial in mathematics and sciences but also play a pivotal role in human cognitive processes (Román-González et al., 2017; Shute et al., 2017).

From the field of cognitive psychology, these skills are related to developing executive functions such as planning, cognitive flexibility and working memory, which are vital

factors in mathematical reasoning and fluid intelligence (Diamond, 2013; Robledo-Castro et al., 2023, 2025). In this sense, the meta-analysis conducted by Scherer et al. (2019) confirms the educational impact of IT programming on the logical-mathematical reasoning of high school students.

Barcelos et al. (2018) highlight that the impact of computational thinking in mathematics is linked to the development of higher-order cognitive skills, such as identifying patterns and building mathematical models, thus furthering interest in including computational thinking in the high school curriculum.

The use of IT programming environments such as Scratch and Python can thus prove effective when learning mathematical concepts. According to De La Hoz and Hijón (2022), the use of block programming environments such as Scratch helps with the learning of mathematical concepts and improves both academic performance and attitudes toward mathematics. Basogain-Olabe et al. (2015) add that the use of Scratch improves the problem-solving described in the PISA.

Python has also proved to be effective when learning mathematics by helping students to conceptualize complex mathematical structures. Pinargote-Zambrano et al. (2024) found that using it enabled higher education students to enhance their understanding of first- and second-degree equations. Zambrano-Choez et al. (2025) conclude that computational thinking and programming—when applied to learning mathematical concepts—can strengthen critical thinking and higher-level cognitive skills such as logical reasoning, problem solving, and spatial visualization.

Authors such as Moreno-Léon et al. (2017) point out that the gradual transition from programming languages through blocks toward languages that require the writing of code can offer an optimal pathway by which to consolidate computational skills, since students move on from what are highly accessible environments to more complex programming structures.

As regards transferring computational thinking to logical-mathematical reasoning, Scherer et al. (2019) evidenced the impact of far transfer and distinguished between near transfer ($g$ = 0.75) and far transfer ($g$ = 0.47), highlighting that the success of these activities depends on the educational context and on how the intervention is designed.

Nevertheless, there are obstacles to integrating computational thinking into mathematics. Prominent amongst these hurdles are: (a) the scant training and experience of teachers in the area (Tariq et al., 2024), and (b) the difficulty involved in solving complex problems using programming tools, which might discourage learning and even lead to students giving up trying to learn how to program (da Silva et al., 2025).

From a mathematics education perspective, computational thinking should not be understood as learning to code per se, but rather as a set of cognitive tools that support core mathematical reasoning processes, including abstraction, modeling, generalization, and structured problem solving.

### *1.3. Model of Computational Thinking and Cognitive Development*

From a cognitive perspective, there are several models that link computational thinking to logical-mathematical reasoning. According to Brennan and Resnick (2012), computational practices such as iteration, debugging, and abstraction trigger mental processes related to problem solving in a structured manner. Indeed, Shute et al. (2017) view computational thinking as a set of high-level cognitive skills.

The systematic review and meta-analysis recently conducted by Montuori et al. (2024) sums up the link between computational thinking and cognitive development. The analysis carried out in the 19 experimental studies into what impact computational thinking has on the executive functions of children and teenagers shows significant effects on problem

solving ($d$ = 0.89), planning ($d$ = 0.36), inhibition, and working memory. The authors also state that student age acts as a moderating variable, such that non-structured visual programming activities, particularly with Scratch, prove to be more efficient in older children.

### 1.4. Empirical Evidence of the Impact of Computational Thinking by Education Level

Empirical evidence has shown that the impact of computational thinking varies depending on the stage of education involved. In compulsory secondary education from 12 to 16 years of age (Spanish acronym: ESO), significant improvements were observed in variables related to numerical reasoning, geometry, and problem solving after intervention with IT programming (Kalelioğlu et al., 2016; Sáez-López et al., 2016; Xu et al., 2022). In a similar vein, the meta-analysis carried out by Li and Oon (2024) finds that the stage of education moderates the impact of computational thinking, with greater effect sizes in the early stages compared to a gradual drop as students move up toward higher stages of the education system. Viewed from the field of developmental psychology, this pattern may be interpreted through the change from concrete thinking to formal thinking described by Piaget (1981), as well as through Vygotsky's concept of the zone of proximal development (Vygotsky et al., 1978), which accounts for a greater margin for improvement in the early stages of secondary education.

In this context, the present study seeks to assess what impact educational intervention based on IT programming activities using Scratch 2.0 (MIT Media Lab, Cambridge, MA, USA), and Python 3.8 (Python Software Foundation, Beaverton, OR, USA) can have on the logical-mathematical reasoning of high school students. The aim is also to examine if improvements vary depending on the stage of education—from 12-year-olds to 17-year-olds—and in terms of the groups (comparison and experimental). Finally, a further goal is to explore the principal effects of the group and level of education, as well as the interaction between the two, in order to pinpoint in which school years intervention has proved most effective and whether its impact remains consistent throughout the various stages of education.

## 2. Materials and Methods

### 2.1. Participants

The study involved 388 students from two schools: a public high school in the province of Valladolid (Spain) and a Spanish state-run school in Bogota (Colombia), which follows the same Spanish educational system and official curriculum as schools in Spain. Student distribution by stage of education and group is shown in Table 1.

**Table 1.** Distribution of the sample by education level and group.

| | **Total** | **Level** | | | | | | **Group** | |
|---|---|---|---|---|---|---|---|---|---|
| | | **1st ESO** | **2nd ESO** | **3rd ESO** | **4th ESO** | **1st Bach** | **2nd Bach** | **Comparison** | **Experimental** |
| Frequency | $N$ = 388 | $n$ = 90 | $n$ = 89 | $n$ = 56 | $n$ = 58 | $n$ = 43 | $n$ = 52 | $n$ = 186 | $n$ = 202 |
| % | 100% | 23.2% | 22.9% | 14.4% | 14.9% | 11.1% | 13.4% | 47.9% | 52.1% |

ESO = Compulsory Secondary Education (12–15-year-olds); Bach = *Bachillerato* (Spanish Upper Secondary, 16–17-year-olds). $n$ = sample size. Percentages are calculated out of the total sample ($N$ = 388).

### 2.2. Instruments

In order to assess general cognitive aptitudes and, in particular, those related to logical-mathematical reasoning, the BADyG/M-r (Differential and General Skills Battery, middle level, updated version) psychometric battery was applied for years one to four of the ESO, and BADyG/S-r (Differential and General Skills Battery, high level, updated version) for years one and two of *Bachillerato*/Upper Secondary.

BADyG (Yuste & Martínez, 2003a, 2003b) is one of the most widely applied tools in education and educational psychology thanks to its high degree of validity and reliability (Monsalvo Díez & Carbonero Martín, 2009). It is an intelligence test—seen as the capacity to undertake activities that require abstract thinking and the ability to predict possible consequences of hypothetical situations.

This construct is assessed through six factors: (a) verbal analogies, (b) numerical series, (c) logical matrices, (d) sentence completion, (e) numerical problems, and (f) matching shapes. These factors are grouped into others that are second order: (a) logical reasoning, which includes verbal analogies, numerical series, and logical matrices; (b) verbal factor, which encompasses verbal analogies and sentence completion; (c) numerical factor, comprising numerical series and numerical problems; and (d) visual-spatial factor, which covers the variables of logical matrices and matching shapes. The sum of these four complementary factors provides a global value—general intelligence.

#### 2.2.1. Validity of the Construct

Official assessments of the BADyG have provided ample evidence of its validity based on the internal structure. In the case of the BADyG/M-r, exploratory factorial analyses show a general intelligence factor and three correlated factors (verbal, numerical, and visual-spatial), which is consistent with the battery's theoretical framework.

These results have also been shown to support the hierarchical conception of intelligence and factorial organization proposed by the authors. Likewise, in the BADyG/S-r, principal component analyses have confirmed the presence of a general factor and of the specific factors that make up the test, thus offering sufficient evidence of the construct's validity (Spanish General Council of Psychology, 2019a, 2019b).

#### 2.2.2. Concurrent Validity

Both versions of the psychometric battery provide empirical evidence of validity based on the link to relevant external criteria. In BADyG/M-r, significant correlations have been reported to academic performance in language, mathematics, English, and social sciences, with values deemed appropriate for assessing cognitive aptitudes amongst schoolchildren.

These relations are predictive, since the BADyG is applied at the start of the school year, with scores then being reported when the school year ends. As regards the BADyG/S-r, positive correlations have also been found with academic performance, with mean values of between 0.20 and 0.35, thereby offering further evidence of validity in secondary students aged 16 to 17 (Spanish General Council of Psychology, 2019a, 2019b).

#### 2.2.3. Reliability

Indices of internal consistency to assess the reliability of the scores have been analyzed through the Cronbach alpha coefficient and the split-half corrected method. In the BADyG/M-r, values showed coefficients above 0.75 in the basic tests, over 0.90 in the global factors, and over 0.94 in general intelligence (Spanish General Council of Psychology, 2019a).

In the case of the BADyG/S-r, internal consistency varies between 0.70 and 0.86 in the basic tests, and between 0.83 and 0.88 in the global factors, reaching values of between 0.93 and 0.94 in general intelligence (Spanish General Council of Psychology, 2019b). These results confirm the psychometric robustness of the two batteries for use in research in education.

### *2.3. Methodological Design and Procedure*

Since the participating groups were already formed in each school, and because no random allocation could be made, the study was organized with a quasi-experimental design that included a non-equivalent comparison group, which is common in research

in education in natural environments. Work was carried out with an experimental group, whose activities focused on computational thinking, and a comparison group, which proceeded with their regular curriculum.

In both educational contexts, existing class groups were assigned either to the experimental group or to the comparison group within each school; class assignment to condition was conducted at the intact class-group level, based on organizational and timetable constraints, in order to avoid disruption of regular school functioning and to minimize contamination between groups. Thus, both schools included experimental and comparison groups following the same institutional framework. All intervention activities were implemented during regular class time within the subject "Technology and Digitalization". No additional instructional time or extracurricular sessions were introduced for the experimental group. This design reduces the likelihood that observed effects are attributable to increased instructional time or general enrichment, rather than to the specific characteristics of the intervention.

In order to ensure the study's internal validity, a pretest ($O_1$ and $O_3$) and a post-test ($O_2$ and $O_4$) were applied to the two groups, which enabled a comparison to be made not only with regard to the initial differences between them but also in terms of improvements achieved in the study variables, as suggested in classical quasi-experimental designs (Creswell & Creswell, 2021; Mertens, 2020). Below is the outline of the quasi-experimental design used in the study (Table 2).

**Table 2.** Outline of the quasi-experimental design carried out in the study.

| | Pre-Test | Intervention | Post-Test |
|---|---|---|---|
| Experimental group | $O_1$ | X | $O_2$ |
| Comparison group | $O_3$ | – | $O_4$ |

$O_1$ and $O_3$: pretest measure; $O_2$ and $O_4$: post-test measure; X: intervention applied; –: no intervention applied.

The study was carried out in two schools, one in the province of Valladolid and another in a Spanish state-run school in Bogota (Colombia). Both schools operate under the same Spanish educational system and official curriculum, ensuring curricular equivalence across contexts. The methodological sequence, materials and intervention conditions were identical in the two schools, which allowed the results to be merged into a single comparative analysis.

The intervention was implemented by the same research team in both contexts, following a shared instructional plan and common progression criteria, in order to ensure consistency across groups. In the two schools, intervention lasted for about six months and was organized in the same way: 40 sessions of 50 min each, undertaken during school time. In the high school in Valladolid (Spain), intervention took place over one school year, applying the BADyG battery as a pre-test halfway through the first term (October) and as a post-test prior to the end of the school year (May). In the school in Bogota (Colombia), the same procedure was employed in the following school year, with the BADyG also being applied at the start and at the end of the intervention in the same months. The study was therefore conducted over two consecutive academic years, and all experimental and comparison groups from both cohorts were included in the statistical analyses once baseline equivalence had been verified through pretest scores.

The design of the intervention program involved carrying out activities that gradually increased in complexity. Solving the activities initially required basic algorithms, after which more advanced programming structures had to be tackled. All the computer programs included mathematical concepts (mental calculation, algebra, geometry, and mathematical analysis) that needed to be solved. These activities were designed using

a common set of tasks and materials that were implemented uniformly in both schools, without contextual adaptations, in order to ensure internal consistency of the intervention.

The intervention was explicitly designed to promote core computational thinking practices—such as decomposition, algorithmic reasoning, abstraction, and debugging—rather than general digital skills or generic enrichment activities.

Students in the comparison group attended the same subject during the same time slots but followed the official curriculum of the subject "Technology and Digitalization", without engaging in programming or computational thinking activities.

In order to help students delve deeper into the concepts, practical tasks and creative activities were systematically combined.

- Stage 1: All the students in the experimental group undertook an initial stage of 20 sessions using Scratch to reduce the cognitive load that is often associated with text programming languages and to avoid initial rejection that might arise due to syntaxis and level of abstraction (Tsai et al., 2025). The Scratch environment allowed basic computational concepts such as sequences, loops, conditionals, parallelism, and debugging to be more accessible to students. Graphical representations and flow diagrams were also used to encourage algorithmic thinking.
- Stage 2: Students from the experimental group of first, second, and third year ESO students (12–15-year-olds) completed the whole of the intervention using only Scratch up to the 40 planned sessions. Nevertheless, once the bases had been established, fourth-year ESO and upper-secondary students worked for a further 20 sessions with Python, thus completing the intervention.

Python is a programming language that is more appropriate for developing complex structures and enables an easier transition to more advanced mathematical contexts. Once students had become more familiar with the environment, they gradually engaged in increasingly more complex activities with different kinds of data, mathematical operations, lists, tuples, and libraries. Exercises were also carried out with variables, conditional structures (if/else), loop control structures (while, for, until) and programs in which functions needed to be defined. The use of standard modules such as math, time, and random was also introduced. Some of the main programs undertaken by students involved creating passwords, counters, timers, guessing games, and "hangman". These activities were designed to help overcome the difficulties that those who are just beginning to program tend to face when moving up from block to text programming environments.

According to Mladenović et al. (2024), the transition from block languages to text languages can lead to errors in conception and can pose cognitive barriers that must be dealt with if a deeper understanding of programming is to be achieved. All sessions were delivered following the same instructional guidelines and progression criteria in both educational contexts, allowing any observed differences in outcomes to be attributed to the intervention rather than to differences in task design or implementation.

Although no formal external fidelity checks (e.g., observational protocols or fidelity checklists) were applied, the use of a common instructional framework and shared implementation criteria contributed to maintaining instructional consistency across contexts.

### 2.4. Data Analysis

Once the assumptions for the parametric tests had been verified, a descriptive statistical analysis was carried out, followed by inferential analysis, to gauge what impact the intervention had. In order to compare the results between the experimental group and the comparison group, gains (differences between post- and pre-test) were considered in all the variables of the BADyG, after verifying initial equivalence between the two groups in each level of education through the pre-test scores.

Although analysis of covariance (ANCOVA) is frequently used in pretest–posttest designs, it was not applied in the present study. In quasi-experimental designs with non-equivalent groups, the use of pretest scores as covariates may violate the assumption of independence between the covariate and the experimental condition, particularly when initial differences between groups exist, which can lead to biased estimates of the treatment effect (Jamieson, 2004; Miller & Chapman, 2001). For this reason, gain scores were considered the most appropriate analytical strategy in this context, as recommended for quasi-experimental designs with non-equivalent groups.

A two-factor multivariate factorial design was then applied (MANOVA), using education level and group (comparison/experimental) as independent variables, and the difference scores of the complementary factors of the BADyG as dependent variables, following the recommendations of Tabachnick and Fidell (2019) for multivariate analysis in research in education. Effect size was calculated with the partial eta squared ($\eta^2p$), and its values were interpreted following Cohen (1988): $0.01 < \eta^2p < 0.05$, indicating a small effect; $0.06 < \eta^2p < 0.13$, a moderate effect; and $\eta^2p > 0.14$, a large effect.

Post hoc tests were subsequently carried out to identify significant differences between the various levels of education. In our case, we opted for the Scheffé procedure, given its conservative nature as well as its robustness in contexts with unequal sample sizes between education levels (Field, 2018). When comparing the groups, we also applied the *t*-test for independent samples, and incorporated Hedges *g* statistic (Hedges, 1981; Hedges & Olkin, 1985) to calculate effect size, which proves particularly useful in small or unbalanced samples. Interpretation was based on the criteria of Cohen (1988, 1992): (a) $g = 0.20$, indicating a small effect size; (b) $g = 0.50$, a moderate effect size; and (c) $g = 0.80$, representing a large effect size. All the analyses were conducted with a 95% confidence level using the IBM SPSS Statistics v.29 (IBM Corp., Armonk, NY, USA) statistical package.

Since the research was conducted at two different schools and covered two consecutive school years, group equivalence was verified prior to intervention through pre-test comparisons. This procedure reduces the likelihood that post-test differences are explained by baseline variability between groups and supports a more cautious interpretation of observed gains as being associated with the computational thinking intervention under the conditions of this study.

## 3. Results

### *3.1. Descriptive Analysis*

In line with the results in Table 3, the descriptive analysis shows that gains in the complementary factors of the BADyG are greater in compulsory secondary (ESO) than in upper secondary (*Bachillerato*). In particular, first and second year ESO students stand out for the greatest improvements in most factors, whereas these gains tend to diminish as students progress through later years, with the lowest values being achieved in *Bachillerato*. This trend is particularly clear in the numerical reasoning factor, the visual-spatial reasoning factor, and in general intelligence, where the differences between the levels are more marked. In contrast, the verbal factor displayed minimal variations between school years, suggesting general stability in this aspect.

When comparing the groups, an even clearer pattern emerges; the experimental group obtains systematically higher gains than the comparison group in all factors, except in the verbal factor, where the two groups show very small increases. Particularly noteworthy are the major differences in the visual-spatial factor and in general intelligence, in which the improvements in the experimental group were markedly greater.

**Table 3.** Distribution and descriptive statistics of the (post-pre) gain scores in the complementary factors of BADyG in terms of education level and group.

| | Total | Level | | | | | | Group | |
|---|---|---|---|---|---|---|---|---|---|
| | | 1st ESO | 2nd ESO | 3rd ESO | 4th ESO | 1st Bach | 2nd Bach | Comparison | Experimental |
| Frequency | $N$ = 388 | $n$ = 90 | $n$ = 89 | $n$ = 56 | $n$ = 58 | $n$ = 43 | $n$ = 52 | $n$ = 186 | $n$ = 202 |
| % | 100% | 23.2% | 22.9% | 14.4% | 14.9% | 11.1% | 13.4% | 47.9% | 52.1% |
| dif_RR | | | | | | | | | |
| M | 3.45 | 3.43 | 5.00 | 3.27 | 3.19 | 2.98 | 1.71 | 1.06 | 5.65 |
| SD | 6.09 | 5.90 | 5.43 | 6.24 | 5.74 | 7.54 | 6.02 | 5.31 | 5.94 |
| dif_VV | | | | | | | | | |
| M | 0.79 | 0.68 | 1.30 | 1.05 | 1.21 | 0.67 | −0.56 | 0.64 | 0.93 |
| SD | 4.79 | 4.47 | 4.55 | 4.36 | 5.29 | 6.16 | 4.18 | 4.60 | 4.95 |
| dif_NN | | | | | | | | | |
| M | 3.55 | 3.98 | 4.78 | 2.46 | 2.66 | 3.79 | 2.69 | 1.36 | 5.57 |
| SD | 4.67 | 4.80 | 5.03 | 4.50 | 4.07 | 4.45 | 4.36 | 3.82 | 4.48 |
| dif_VE | | | | | | | | | |
| M | 3.40 | 4.86 | 4.33 | 2.89 | 3.02 | 1.35 | 2.00 | 0.87 | 5.74 |
| SD | 4.83 | 5.40 | 4.74 | 5.14 | 3.62 | 4.03 | 4.40 | 3.91 | 4.40 |
| dif_IG | | | | | | | | | |
| M | 7.74 | 9.51 | 10.40 | 6.41 | 6.88 | 5.81 | 4.13 | 2.87 | 12.24 |
| SD | 9.37 | 9.94 | 9.07 | 9.37 | 8.07 | 9.74 | 8.27 | 7.69 | 8.50 |

M = mean; SD = standard deviation; dif_RR = post-pre difference in logical reasoning; dif_VV = post-pre difference in verbal factor; dif_NN = post-pre difference in numerical factor; dif_VE = post-pre difference in visual-spatial factor; dif_IG = post-pre difference in general intelligence.

Taken as a whole, the descriptive results suggest that intervention based on computational thinking leads to greater gains in the cognitive skills assessed, with a more marked impact in the early years of the ESO, and a more moderate impact in *Bachillerato*. Descriptively, the clearest and most consistent patterns of improvement are observed in the numerical, visual-spatial, and general intelligence factors. Furthermore, the advantage gained by the experimental group in virtually all the factors anticipates a possible positive effect of the intervention, an aspect that needs to be tested through inferential analysis.

### *3.2. Multivariate Analysis*

In order to determine how the effects of education level and group affect the dependent variables related to the post-pre difference scores in the complementary factors of the BADyG, a multivariate analysis of variance (MANOVA) was carried out using a 6 × 2 design. Prior to conducting the MANOVA, the assumptions of multivariate analysis were examined (e.g., linearity and absence of multicollinearity). Box's M test was used to assess the homogeneity of covariance matrices across groups. The results indicated no substantial deviations from the assumptions; therefore, Wilks' lambda was employed as the principal statistic, given its widespread use and robustness to minor assumption violations. The stability of the findings was further confirmed using alternative multivariate statistics (Pillai's trace, Hotelling–Lawley trace, and Roy's largest root), following methodological recommendations for multivariate analysis (Field, 2018).

The MANOVA results shown in Table 4 reveal significant principal effects for education level and group, albeit with different-sized effects. The education level factor showed a significant effect, Wilks' $\Lambda = 0.845$, $F(20, 1238) = 3.152$, $p < 0.001$, $\eta^2p = 0.041$, suggesting there are multivariate differences between school years with regard to the improvements obtained. Nevertheless, this effect is small, which is consistent with the diversity of the education levels involved in the study.

The principal effect of the group also proved significant—Wilks' $\Lambda = 0.664$, $F(4, 373) = 47.13$, $p < 0.001$, $\eta^2p = 0.336$—indicating a large effect size. Consequently, these results generally indicate that the students in the experimental group displayed significantly higher gains and improvements in the variables studied when compared to the comparison

group. As regards the interaction between level and group, the results show no significant differences: Wilks' $\Lambda = 0.920$, $F(20, 1238) = 1.57$, $p = 0.052$, $\eta^2p = 0.021$.

**Table 4.** Multivariate analysis of variance (6 [a] × 2 [b]).

| | Wilks' Λ | *F* | *p* | $\eta^2p$ |
|---|---|---|---|---|
| Level [a] | 0.845 | $F(20, 1238) = 3.22$ | <0.001 | 0.041 |
| Group [b] | 0.664 | $F(4, 373) = 47.13$ | <0.001 | 0.336 |
| Level × Group | 0.920 | $F(20, 1238) = 1.57$ | 0.052 | 0.021 |

Λ = Wilks Lambda; *F* = Snedecor F; *p* = level of significance; $\eta^2p$ = partial eta squared. [a] Level of education $a_1$ = 1st ESO, $a_2$ = 2nd ESO, $a_3$ = 3rd ESO, $a_4$ = 4th ESO, $a_5$ = 1st *Bachillerato*, $a_6$ = 2nd *Bachillerato*. [b] Group: $b_1$ = Comparison; $b_2$ = Experimental.

In other words, although the tendency shows values close to the significance thresholds ($\alpha = 0.05$), differences between groups do not vary significantly in terms of the school year.

### *3.3. Principal Effects of the Level of Education Variable*

The results of the univariate analyses showed statistically significant differences between education levels in three of the complementary factors of the BADyG (Table 5), all with small size effects, which is very common in multi-level education studies. Notably, the visual-spatial factor and general intelligence show the clearest developmental pattern across school years. Significant differences were found in: (a) numerical factor, $F(5, 382) = 3.397$, $p = 0.005$; (b) visual-spatial factor, $F(5, 382) = 6.973$, $p < 0.001$; and (c) general intelligence, $F(5, 382) = 5.733$, $p < 0.001$.

**Table 5.** Means, standard deviation, *F* values, levels of significance (*p*), effect size ($\eta^2p$) and post hoc comparisons (Scheffé) of the difference scores of the BADyG by level of education.

| Level | dif_RR M (SD) | dif_VV M (SD) | dif_NN M (SD) | dif_VE M (SD) | dif_IG M (SD) |
|---|---|---|---|---|---|
| 1st ESO [a] | 3.43 (5.90) | 0.68 (4.47) | 3.98 (4.80) | 4.86 (5.39) | 9.51 (9.94) |
| 2nd ESO [b] | 5.00 (5.42) | 1.30 (4.55) | 4.78 (5.03) | 4.33 (4.73) | 10.40 (9.06) |
| 3rd ESO [c] | 3.27 (6.24) | 1.05 (4.36) | 2.46 (4.50) | 2.89 (5.14) | 6.41 (9.37) |
| 4th ESO [d] | 3.19 (5.73) | 1.21 (5.28) | 2.66 (4.06) | 3.02 (3.62) | 6.88 (8.07) |
| 1st Bach [e] | 2.98 (7.54) | 0.67 (6.16) | 3.79 (4.45) | 1.35 (4.03) | 5.81 (9.74) |
| 2nd Bach [f] | 1.71 (6.02) | −0.56 (4.18) | 2.69 (4.36) | 2.00 (4.39) | 4.13 (8.27) |
| *F*(5, 382) | 2.229 | 1.140 | 3.397 | 6.973 | 5.733 |
| *p* | 0.051 | 0.339 | 0.005 | <0.001 | <0.001 |
| $\eta^2p$ | — | — | 0.043 | 0.085 | 0.071 |
| Post hoc (Scheffé) | — | — | No significant differences observed | 1st ESO > 1st Bach; 1st ESO > 2nd Bach; 2nd ESO > 1st Bach; 2nd ESO > 2nd Bach | 1st ESO > 2nd Bach; 2nd ESO > 2nd Bach |

M = mean; SD = standard deviation; *F* = Snedecor F; *p* = level of significance; $\eta^2p$ = partial eta squared. dif_RR = post-pre difference in logical reasoning factor, dif_VV = post-pre difference in verbal factor, dif_NN = post-pre difference in numerical factor, dif_VE = post-pre difference in visual-spatial factor, dif_IG = post-pre difference in general intelligence. [a] $n = 90$, [b] $n = 89$, [c] $n = 56$, [d] $n = 58$, [e] $n = 43$, [f] $n = 52$.

In these three variables, the highest scores were reported in the first and second year ESO, whereas scores in later school years tended to drop gradually, particularly in *Bachillerato* (Figure 1). As a result—and according to this pattern—improvements tend to be more significant in the early years of secondary education.

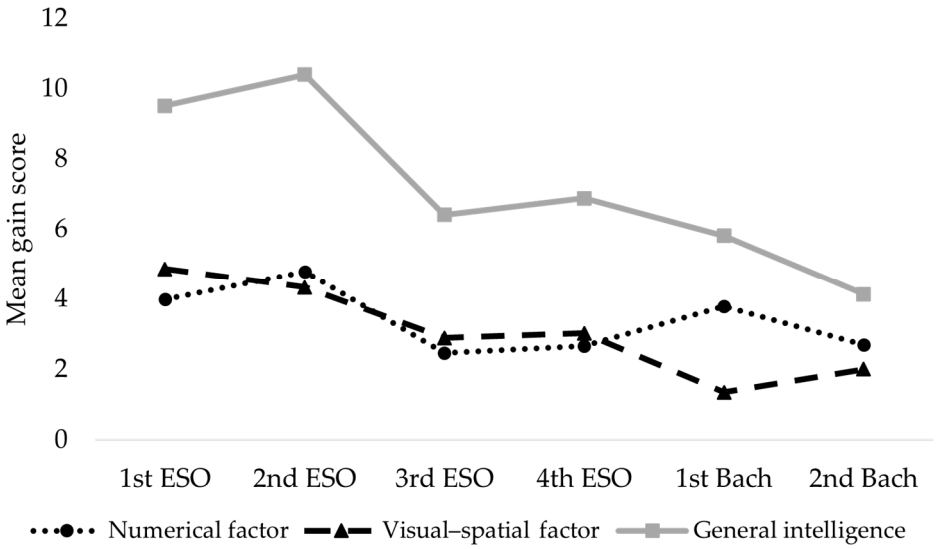



**Figure 1.** Mean gain scores (post-pre) in numerical, visual-spatial, and general intelligence factors by educational level in the experimental group.

Variance analysis showed significant differences between education levels in the numerical, visual-spatial, and general intelligence factors. Depending on the $\eta^2 p$ obtained, effect sizes varied between small and moderate. Post hoc comparisons carried out using the Scheffé procedure allowed us to identify specific differences between education levels. Specifically, with regard to the visual-spatial factor, first- and second-year ESO students obtained significantly higher gains than those in the two years of *Bachillerato*. For general intelligence, significant differences were found between first and second year ESO and second year *Bachillerato*. Finally, no significant post hoc differences were observed for the numerical factor despite the significant main effect of education level.

In contrast, no significant differences were found between levels in the logical reasoning factor, $F(5, 382) = 2.229$, $p = 0.051$ or verbal factors, $F(5, 382) = 1.140$, $p = 0.339$, indicating that improvements in these areas are more evenly distributed over the different school years.

In sum, these results indicate that education level has a significant impact on improvements in the factors of numerical reasoning, visual-spatial reasoning, and general intelligence, whereas gains in logical reasoning and the verbal factor do not depend on the school year.

### *3.4. Principal Effects of the Group Variable*

The results shown in Table 6 show statistically significant differences between groups in four of the five complementary factors of the BADyG, all with effect sizes ranging from moderate to large. The largest effects were observed in the numerical, visual-spatial, and general intelligence factors.

As shown in Figure 2, the experimental group achieved substantially higher gain scores than the comparison group in logical reasoning, numerical reasoning, visual-spatial reasoning, and general intelligence.

Overall, the experimental group achieved post-pre gains that were significantly higher than those of the comparison group in logical reasoning, numerical reasoning, visual-spatial reasoning, and general intelligence, with effect sizes ranging from moderate to large (Table 6).

**Table 6.** Means (standard deviations), *t* values, levels of significance (*p*) and effect sizes (Hedges *g*) for gains in the complementary factors of the BADyG in terms of the group.

| | Comparison [a] M (SD) | Experimental [b] M (SD) | *t*(386) | *p* | *g* |
|---|---|---|---|---|---|
| dif_RR | 1.06 (5.31) | 5.65 (5.94) | −7.99 | <0.001 | 0.81 |
| dif_VV | 0.64 (4.61) | 0.93 (4.95) | −0.59 | 0.557 | 0.06 |
| dif_NN | 1.36 (3.82) | 5.57 (4.48) | −9.92 | <0.001 | 1.01 |
| dif_VE | 0.87 (3.91) | 5.74 (4.40) | −11.51 | <0.001 | 1.17 |
| dif_IG | 2.87 (7.69) | 12.24 (8.50) | −11.36 | <0.001 | 1.15 |

M = mean; SD = standard deviation; *t* = Student t; *p* = level of significance (α = 0.05); *g* = Hedges' effect size; dif_RR = post-pre difference in logical reasoning, dif_VV = post-pre difference in verbal factor, dif_NN = post-pre difference in numerical factor, dif_VE = post-pre difference in visual-spatial factor, dif_IG = post-pre difference in general intelligence. [a] *n* = 186, [b] *n* = 202.

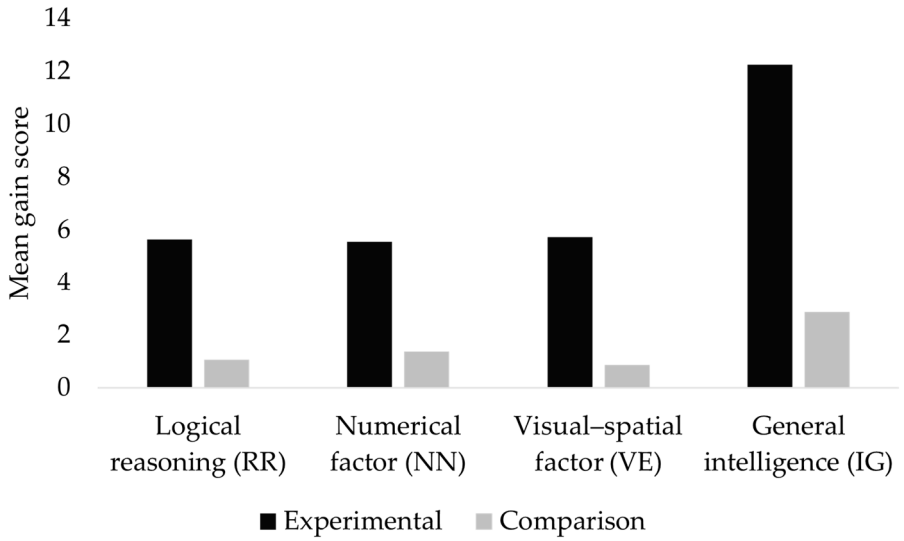



**Figure 2.** Mean gain scores (post-pre) in the BADyG complementary factors for the experimental and comparison groups.

The verbal factor showed no significant differences between groups, $t(386) = -0.59$, $p = 0.557$, with a very small effect size ($g = 0.06$), which suggests that improvements in this area were similar in the two groups.

In sum, these results confirm that intervention based on computational thinking yielded improvements in most of the cognitive factors assessed, with particularly marked effects in numerical, visual-spatial, and general intelligence factors.

### *3.5. Effects of Interaction Between Education Level and Group*

The analyses showed no significant effect of interaction between education level and group (comparison and experimental) on the difference scores of the BADyG—Wilks' $\Lambda = 0.920$, $F(20, 1238) = 1.572$, $p = 0.052$. Even though the level of significance is close to the threshold, the statistical evidence is insufficient to state that the effect of the intervention clearly varies between the different levels of education.

This trend indicates that there might be slight variations in the scale of the improvements between school years, but that said, differences are not sufficiently consistent to consider them as offering a solid pattern. As a result, no additional analyses of simple effects by level were carried out, since the significance of the interaction is a prior requirement to interpret specific comparisons within each school year.

Overall, the results indicate that no statistically significant differential effects of the intervention were observed across educational levels, with gain patterns remaining broadly comparable throughout secondary education.

## 4. Discussion

The results of this study indicate that high school students who take part in an educational program that involves programming with Scratch and Python—designed to promote computational thinking—obtain significant improvements in several dimensions of logical-mathematical reasoning, assessed using the BADyG psychometric battery.

In particular, gains in these variables in the experimental group were higher than those in the comparison group in logical reasoning, numerical factor, visual-spatial factor, and general intelligence, with effect sizes that range from moderate to large. However, no significant differences were found in the verbal factor between groups. Given the nature of the intervention, which primarily emphasizes logical-mathematical reasoning and problem-solving processes, substantial changes in verbal intelligence were not necessarily expected. The absence of significant differences in the verbal factor is therefore consistent with the cognitive focus of computational thinking-based activities.

In this context, gains observed in the general intelligence factor should be interpreted as improvements in test-related cognitive performance sensitive to educational intervention, rather than as changes in stable or trait-like intelligence constructs.

From an educational standpoint, the strongest and most meaningful effects were observed in the numerical, visual-spatial, and general intelligence factors, suggesting that computational thinking interventions particularly reinforce core cognitive processes underlying logical-mathematical reasoning rather than producing generalized gains across all cognitive domains.

Furthermore, education level had a significant principal effect, with greater gains in the early years of ESO followed by a tendency to gradually diminish in *Bachillerato*, although it was not possible to confirm a statistically significant interaction between level and group. As affirmed by Li and Oon (2024), this pattern is consistent with the role played by education level as a moderating variable in the impact of computational thinking.

These results concur with other studies that link computational thinking to the development of advanced cognitive skills, particularly those related to logical-mathematical reasoning (Grover & Pea, 2013; Scherer et al., 2019; Shute et al., 2017; Zambrano-Choez et al., 2025). In line with Wing (2008) as well as the ISTE and CSTA (2011), computational thinking encompasses processes such as decomposition, abstraction, the use of algorithms, pattern identification, and generalization, all of which tie in directly with the ability to solve mathematical problems in a structured and efficient manner.

In this context, improvements observed in numerical factor, visual-spatial factor, and general intelligence may be seen as boosting the logical organization of thinking, symbolic representation and problem modeling, as highlighted by Román-González et al. (2017).

The results show a pattern with a significant impact on the numerical and visual-spatial factors. Using environments such as Scratch and Python enabled students to work with concepts like coordinates, variables, control structures, graphical representations, and algorithmic models to solve problems. This, in turn, helped with the understanding of abstract mathematical concepts and the exploration of the logical underlying structures (Pinargote-Zambrano et al., 2024; Sáez-López et al., 2016). From a constructionist perspective (Papert, 1980, 1993), creating one's own projects in these environments becomes a driver for learning, since students "program to learn" rather than merely "learning to program", and thus build mathematical knowledge by elaborating and experimenting with artifacts that are meaningful both to the learners and to their learning context.

As regards the effect of education level, results indicate that improvements in numerical reasoning, visual-spatial reasoning, and general intelligence occur primarily in the early years of ESO. As students progress through the education system, these gains become more modest, especially in *Bachillerato*. This may be explained from the standpoint of developmental psychology; the transition from concrete operations toward the more formal thinking described by Piaget (1981), as well as the role played by the zone of proximal development in the acquisition of new cognitive skills (Vygotsky et al., 1978). It is reasonable to assume that there is a greater margin for improvement in the early years and that tasks based on computational thinking adapt best to students' zone of proximal development. In contrast, in *Bachillerato*, many of these cognitive structures have already been well established, which might account for there being less of an improvement. Such reasoning concurs with the meta-analysis conducted by Li and Oon (2024), who point out that education level moderates the impact of computational thinking, with stronger effects at earlier educational stages.

In this regard, the absence of a statistically significant interaction between education level and group suggests that, while the magnitude of gains varies across school years, the intervention does not produce qualitatively different effects at specific educational stages. The observed impact should therefore be interpreted as broadly consistent across secondary education under the instructional conditions of this study.

Use of the BADyG battery as an assessment tool fully aligns with these results, as it allows for the measurement of different aptitudes such as numerical series, logical matrices, numerical problems or matching shapes whilst also grouping them into second-order factors. This offers a clearer and more structured representation of intelligence, together with a suitable level of validity and reliability in educational contexts (Yuste & Martínez, 2003a, 2003b). In this study, the fact that improvements were found mainly in the numerical and visual-spatial factors, as well as in general intelligence, aligns with previous works; computational thinking and IT programming boost the processes of logical reasoning that underpin these dimensions (Román-González et al., 2017; Scherer et al., 2019; Shute et al., 2017).

*Limitations and Proposals for Improvement*

Although the results of this study do provide consistent evidence concerning the impact that computational thinking has on logical-mathematical reasoning, certain limitations should be taken into account. First, although intervention was carried out at two different schools (one in Spain and another in Colombia) and over two consecutive academic years, student distribution by education levels was not totally homogeneous. This might have affected the sensitivity of some analyses, particularly in the level × group interaction, where the results showed $p$-values that were very close to the significance threshold. It would prove advisable to repeat this study using samples that are more balanced between school years and between schools in order to enhance the validity of the results.

Additionally, as the study relied on a quasi-experimental design with non-randomized groups, residual initial differences between groups cannot be entirely ruled out. Although gain scores were considered the most appropriate analytical strategy under these conditions, future research employing randomized designs or alternative longitudinal modeling approaches could further strengthen causal inference. In addition, variables not captured by the pretest (e.g., prior academic achievement, teacher effects, classroom climate, or other contextual factors) may have differed between conditions and could have influenced the observed outcomes.

Secondly, the intervention was carried out by the researchers themselves. This might have led to bias, either due to prior expectations or due to the manner in which ac-

tivities were conducted. In order to offset this possible effect, future research should draw on external teachers when applying the program or should include a double-blind assessment design.

Furthermore, even though conducting the intervention in two schools and in two countries is a key strength, the study was carried out within a very specific institutional framework, which could lessen the applicability of the results to other educational contexts—both national and international—in which there are different organizational or curricular features. Even under a shared official curriculum, cross-national implementation may involve cultural, organizational, or contextual differences (e.g., school routines, available resources, or student background) that were not modeled explicitly and may have influenced the outcomes. As a result, broadening the research into other educational environments would help to boost the generalization of the results.

Finally, it would be interesting to ascertain how permanent the improvements observed in logical reasoning, numerical, and visual-spatial reasoning prove to be. Likewise, the possible integration of mixed methodologies in studies similar to this (such as qualitative registers through interviews, learning diaries or observations) might help to understand the cognitive processes that underlie the impact of computational thinking.

In this regard, although the present study is based on a quantitative approach that allows for the objective measurement of gains in logical-mathematical reasoning, future research could benefit from the incorporation of qualitative methodologies. Approaches such as semi-structured interviews, classroom observations, or learning diaries would enable a deeper exploration of students' problem-solving strategies, perceptions, and cognitive processes when engaging in programming activities with Scratch and Python. This complementary perspective would contribute to a more comprehensive understanding of how computational thinking interventions influence learning beyond measurable performance gains.

## 5. Conclusions

The present study provides empirical evidence that a program based on activities that promote computational thinking using Scratch and Python as programming environments has a positive educational impact on certain variables related to logical-mathematical reasoning. In particular, the educational impact has a size effect ranging from moderate to large in the factors of numerical and visual-spatial reasoning as well as general intelligence of the BADyG psychometric battery.

The results indicate that education level influences the magnitude of observed gains, with more pronounced improvements in the early years of ESO and more moderate gains in *Bachillerato*. At the same time, the lack of a statistically significant interaction between education level and group suggests that the intervention operates in a comparable manner across secondary education, without evidence of stage-specific effects.

From an educational standpoint, these findings support the potential value of integrating computational thinking within secondary education curricula, provided that such integration is carefully aligned with curricular objectives, instructional design, and contextual constraints, and that appropriate teacher preparation and instructional fidelity are ensured.

Finally, future enquiry should establish longitudinal designs in order to gain insights into how lasting the impact might be and to deepen current understanding of the cognitive processes involved in computational thinking.

**Author Contributions:** Conceptualization, R.P.P., L.J.M.-A. and M.Á.C.-M.; methodology, R.P.P.; software, R.P.P.; validation, R.P.P., L.J.M.-A. and M.Á.C.-M.; formal analysis, R.P.P.; investigation, R.P.P.; resources, L.J.M.-A. and M.Á.C.-M.; data curation, R.P.P.; writing—original draft preparation,

R.P.P.; writing—review and editing, L.J.M.-A. and M.Á.C.-M.; visualization, R.P.P.; supervision, L.J.M.-A. and M.Á.C.-M.; project administration, R.P.P. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Ethical review and approval were waived for this study, as the educational research conducted was of an ordinary nature, did not involve any intervention beyond normal educational practice, and did not entail any foreseeable risks for the participants. The study was conducted in accordance with the principles of the Declaration of Helsinki and was authorized by the management teams of the participating schools.

**Informed Consent Statement:** The informed written consent of the parents or legal guardians was obtained for all the students who took part.

**Data Availability Statement:** The data used in this study are not publicly available due to ethical restrictions, since no specific consent for their public dissemination was either applied for or obtained. Any requests or applications to access the data should be sent to the corresponding author by email.

**Acknowledgments:** The authors wish to express their gratitude to the management teams and to the teaching staff of the schools that took part for their cooperation and support during the study.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

| | |
|---|---|
| BADyG | Differential and General Skills Battery |
| BADyG/M-r | Differential and General Skills Battery, medium level, updated version |
| BADyG/S-r | Differential and General Skills Battery, higher level, updated version |
| CSTA | Computer Science Teachers Association |
| ESO | Compulsory Secondary Education |
| LOGO | LOGO Programming Language |
| ISTE | International Society for Technology in Education |
| MANOVA | Multivariate Analysis of Variance |
| PISA | Program for International Student Assessment |
| STEM | Science, Technology, Engineering and Mathematics |

## References

Barcelos, T. S., Muñoz, R., Villarroel, R., Merino, E., & Silveira, I. F. (2018). Mathematics learning through computational thinking activities: A systematic literature review. *Journal of Universal Computer Science*, *24*(7), 815–841.

Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12. *ACM Inroads*, *2*(1), 48–54. [CrossRef]

Basogain-Olabe, X., Olabe-Basogain, M. Á., & Olabe-Basogain, J. C. (2015). Pensamiento computacional a través de la programación: Paradigma de aprendizaje. *Revista de Educación a Distancia*, *46*. [CrossRef]

Brennan, K., & Resnick, M. (2012, April 13–17). *New frameworks for studying and assessing the development of computational thinking* [Conference presentation]. Annual Meeting of the American Educational Research Association, Vancouver, BC, Canada.

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences*. Lawrence Erlbaum Associates.

Cohen, J. (1992). A power primer. *Psychological Bulletin*, *112*(1), 155–159. [CrossRef]

Creswell, J. W., & Creswell, J. D. (2021). *Educational research: Planning, conducting, and evaluating quantitative and qualitative research* (7th ed.). Pearson.

da Silva, F. X., de Menezes, C. S., & de Castro Junior, A. N. (2025). Dificuldades dos estudantes na aprendizagem de programação: Uma revisão sistemática da literatura. *Cadernos de Educação Tecnologia e Sociedade*, *18*(3), 994–1009. [CrossRef]

De La Hoz, A., & Hijón, R. (2022). Enseñanza de las matemáticas a través del uso de Scratch (transversalidad STEM). *IE Comunicaciones: Revista Iberoamericana de Informática Educativa*, *36*, 14–34.

Diamond, A. (2013). Executive functions. *Annual Review of Psychology*, *64*, 135–168. [CrossRef]

Field, A. (2018). *Discovering statistics using IBM SPSS statistics* (5th ed.). SAGE Publications.

Grover, S., & Pea, R. (2013). Computational thinking in K-12: A review of the state of the field. *Educational Researcher*, *42*(1), 38–43. [CrossRef]

Hedges, L. V. (1981). Distribution theory for Glass's estimator of effect size and related estimators. *Journal of Educational Statistics*, *6*(2), 107–128. [CrossRef]

Hedges, L. V., & Olkin, I. (1985). *Statistical methods for meta-analysis*. Academic Press.

Hsu, T. C., Chang, S. C., & Hung, Y. T. (2018). How to learn and how to teach computational thinking: Suggestions based on a review of the literature. *Computers & Education*, *126*, 296–310. [CrossRef]

International Society for Technology in Education [ISTE] & Computer Science Teachers Association [CSTA]. (2011). *Operational definition of computational thinking for K–12 education*. Available online: https://cdn.iste.org/www-root/Computational_Thinking_Operational_Definition_ISTE.pdf (accessed on 10 February 2026).

Jamieson, J. (2004). Analysis of covariance (ANCOVA) with difference scores. *International Journal of Psychophysiology*, *52*, 277–283. [CrossRef]

Kalelioğlu, F. (2015). A new way of teaching programming skills to K-12 students: Code.org. *Computers in Human Behavior*, *52*, 200–210. [CrossRef]

Kalelioğlu, F., Gülbahar, Y., & Kukul, V. (2016). A framework for computational thinking based on a systematic research review. *Baltic Journal of Modern Computing*, *4*(3), 583–596.

Kallia, M., van Borkulo, S. P., Drijvers, P., Barendsen, E., & Tolboom, J. (2021). Characterising computational thinking in mathematics education: A literature-informed Delphi study. *Research in Mathematics Education*, *23*(2), 159–187. [CrossRef]

Li, Z., & Oon, P. T. (2024). The transfer effect of computational thinking (CT)-STEM: A systematic literature review and meta-analysis. *International Journal of STEM Education*, *11*(1), 44. [CrossRef]

Lye, S. Y., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior*, *41*, 51–61. [CrossRef]

Mertens, D. M. (2020). *Research and evaluation in education and psychology: Integrating diversity with quantitative, qualitative, and mixed methods* (5th ed.). SAGE Publications.

Miller, G. A., & Chapman, J. P. (2001). Misunderstanding analysis of covariance. *Journal of Abnormal Psychology*, *110*(1), 40–48. [CrossRef]

Mladenović, M., Žanko, Ž., & Zaharija, G. (2024). From blocks to text: Bridging programming misconceptions. *Journal of Educational Computing Research*, *62*(5), 1082–1106. [CrossRef]

Monsalvo Díez, E., & Carbonero Martín, M. Á. (2009). Descripción de los factores medidos por la batería BADyG-M y su estudio como variables de intervención educativa. *Revista Iberoamericana de Educación*, *49*(6), 1–9. [CrossRef]

Montuori, C., Gambarota, F., Altoé, G., & Arfé, B. (2024). The cognitive effects of computational thinking: A systematic review and meta-analytic study. *Computers & Education*, *210*, 104961. [CrossRef]

Moreno-Léon, J., Robles, G., & Román-González, M. (2017). Programar para aprender en educación primaria y secundaria: ¿Qué indica la evidencia empírica sobre este enfoque? *ReVisión*, *10*(2), 45–51.

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books.

Papert, S. (1993). *Mindstorms: Children, computers, and powerful ideas* (2nd ed.). Basic Books.

Piaget, J. (1981). La teoría de Piaget. *Infancia y Aprendizaje*, *4*(1), 13–25. [CrossRef]

Pinargote-Zambrano, J. J., Lino-Calle, V. A., & Vera-Almeida, B. J. (2024). Python en la enseñanza de las Matemáticas para estudiantes de nivelación en educación superior. *MQRInvestigar*, *8*(3), 3966–3989. [CrossRef]

Robledo-Castro, C., Castillo-Ossa, L. F., & Hederich-Martínez, C. (2025). Computational thinking training and its effects on working memory, flexibility, and inhibition: Randomized controlled trial in fifth-grade children. *Journal of Applied Developmental Psychology*, *100*, 101850. [CrossRef]

Robledo-Castro, C., Hederich-Martínez, C., & Castillo-Ossa, L. F. (2023). Cognitive stimulation of executive functions through computational thinking. *Journal of Experimental Child Psychology*, *235*, 105738. [CrossRef]

Román-González, M., Pérez-González, J. C., & Jiménez-Fernández, C. (2017). Which cognitive abilities underlie computational thinking? Criterion validity of the computational thinking test. *Computers in Human Behavior*, *72*, 678–691. [CrossRef]

Sáez-López, J. M., Román-González, M., & Vázquez-Cano, E. (2016). Visual programming languages integrated across the curriculum in elementary school: A two-year case study using Scratch in five schools. *Computers & Education*, *97*, 129–141. [CrossRef]

Scherer, R., Siddiq, F., & Viveros, B. S. (2019). The cognitive benefits of learning computer programming: A meta-analysis of transfer effects. *Journal of Educational Psychology*, *111*(5), 764–792. [CrossRef]

Selby, C. C. (2015). Relationships: Computational thinking, pedagogy of programming, and Bloom's taxonomy. In *Proceedings of the workshop in primary and secondary computing education (WiPSCE 2015)* (pp. 80–87). ACM International Conference Proceeding Series. ACM. [CrossRef]

Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review*, *22*, 142–158. [CrossRef]

Spanish General Council of Psychology. (2019a). *Evaluación del test BADyG/M-r: Batería de actitudes diferenciales y generales, Nivel M, renovado*. Spanish General Council of Psychology.

Spanish General Council of Psychology. (2019b). *Evaluación del test BADyG/S-r: Batería de actitudes diferenciales y generales, Nivel S, renovado*. Spanish General Council of Psychology.

Tabachnick, B. G., & Fidell, L. S. (2019). *Using multivariate statistics* (7th ed.). Pearson.

Tariq, R., Aponte Babines, B. M., Ramirez, J., Alvarez-Icaza, I., & Naseer, F. (2024). Computational thinking in STEM education: Current state-of-the-art and future research directions. *Frontiers in Computer Science*, *6*, 1480404. [CrossRef]

Tsai, M. J., Chien, F. P., Sun, W. T., & Jha, N. K. (2025). How block-based programming supports novice learners' coding comprehension: Evidence from eye-tracking lag-sequential analysis. *Computers & Education*, *239*, 105430. [CrossRef]

Vygotsky, L. S., Cole, M., John-Steiner, V., Scribner, S., & Souberman, E. (1978). *Mind in society: The development of higher psychological processes*. Harvard University Press.

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology*, *25*(1), 127–147. [CrossRef]

Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *366*(1881), 3717–3725. [CrossRef]

Xu, E., Wang, W., & Wang, Q. (2022). A meta-analysis of the effectiveness of programming teaching in promoting K-12 students' computational thinking. *Education and Information Technologies*, *28*(6), 6619–6644. [CrossRef]

Yuste, C., & Martínez, R. (2003a). *Batería de aptitudes diferenciales y generales BADyG-M renovado*. CEPE.

Yuste, C., & Martínez, R. (2003b). *Batería de aptitudes diferenciales y generales BADyG-S renovado*. CEPE.

Zambrano-Choez, A. D. J., Barreto-Mero, B. I., Villafuerte-Villacís, I., & Quevedo-Vázquez, H. A. (2025). The role of programming and computational thinking in the development of cognitive skills in mathematics. *Revista Científica INGENIAR: Ingeniería, Tecnología e Investigación*, *8*(16), 167–184. [CrossRef]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.