education sciences

MDPI

Article

Investigating the Effect of Binary Gender Preferences on Computational Thinking Skills

Rose Niousha [1,*], Daisuke Saito [2], Hironori Washizaki [1] and Yoshiaki Fukazawa [1]

[1] Department of Computer Science and Engineering, Waseda University, Shinjuku-ku, Tokyo 169-8555, Japan; washizaki@waseda.jp (H.W.); fukazawa@waseda.jp (Y.F.)

[2] Department of Business Administration, Takachiho University, Suginami-ku, Tokyo 168-0061, Japan; d.saito@takachiho.ac.jp

* Correspondence: rose_niousha@fuji.waseda.jp

Abstract: The Computer Science industry suffers from a vivid gender gap. To understand this gap, Computational Thinking skills in Computer Science education are analyzed by binary gender roles using block-based programming languages such as Scratch since they are intuitive for beginners. Platforms such as Dr. Scratch, aid learners in improving their coding skills by earning a Computational Thinking score while supporting effective assessments of students' projects and fostering basic computer programming. Although previous studies have examined gender differences using Scratch programs, few have analyzed the Scratch project type's impact on the evaluation process when comparing genders. Herein, the influence of project type is analyzed using instances of 124 (62 male, 62 female) projects on the Scratch website. Initially, projects were categorized based on the user's gender and project type. Hypothetical testing of each case shows that the scoring system has a bias based on the project type. As gender differences appear by project type, the project type may significantly affect the gender gap in Computational Thinking scores. This study demonstrates the importance of incorporating the project type's effect into the Scratch projects' evaluation process when assessing gender differences.

Keywords: computational thinking; gender; K-12; Scratch

check for updates

Citation: Niousha, R.; Saito, D.; Washizaki, H.; Fukazawa, Y. Investigating the Effect of Binary Gender Preferences on Computational Thinking Skills. *Educ. Sci.* **2023**, *13*, 433. https://doi.org/10.3390/educsci13050433

Academic Editors: Amber Simpson, Kelli Paul and Pavneet Kaur Bharaj

Received: 14 March 2023
Revised: 15 April 2023
Accepted: 18 April 2023
Published: 23 April 2023

1. Introduction

Science, technology, engineering, and mathematics (STEM) education fosters analytical thinking and innovation in future generations [1]. Although many young people are encouraged to seek STEM careers, a pronounced gender divide exists in such disciplines. According to UNESCO, only 35% of STEM students in higher education worldwide are female [2]. However, the gender disparity in Computer Science (CS) is even more pronounced. In 2020, women held only 25% of computing and mathematics positions in the U.S. [3]. Due to a viewpoint bias, this condition likely inhibits the development of inclusive and user-friendly technological solutions. Therefore, closing the gender gap in CS is crucial to produce distinctive solutions emerging from various viewpoints.

Due to rapid technological advances, educational institutions are increasingly incorporating CS into their curricula. Governments around the world require educators to teach coding, often as part of the primary school curriculum [4]. One goal of implementing CS courses is to strengthen students' Computational Thinking (CT) skills [5]. CT skills are defined as the ability to solve problems, design systems, and understand human behaviors based on the fundamental concepts of CS [6]. Students with diverse educational backgrounds, not just those pursuing a CS field, display CT skills [7]. Problem solvers apply CT skills in areas beyond the CS realm to create tools that address issues rather than simply relying on existing tools [8].

Understanding students' CT skills allows educators to enhance their CS curriculum. In the CS field, CT is a vague concept. Although CT skills in an educational environment can be measured in various ways, the assessment criteria must be carefully defined via specific assessment tools. One approach is the code analysis of projects utilizing block-based programming languages such as Scratch [9].

Some studies have investigated the role of gender on Scratch projects; however, few have examined the impact of the Scratch project type on the evaluation process when comparing the scores by gender. Herein, we aim to elucidate the influence of traditional gender traits on fundamental CS skills. Our study analyzes public Scratch projects on the Scratch website based on the user's gender and the project type. Owing to the lack of easily identifiable user information on the Scratch web interface, gender is extracted from self-reported gender information. Subsequently, we investigate the influence of gender and project type on CT performance. This study aims to answer the following research questions (RQs):

RQ1. Does gender influence the type of Scratch project selected? This question evaluates how the characteristics of the Scratch project type differ by gender groups by comparing the total performance score and gender.

RQ2. Do certain project types demonstrate higher CT scores? This question examines whether a difference exists in the way projects are scored when gender is not considered.

RQ3. Is there a gender gap in CT skills? This question clarifies whether a difference exists in the CT score based on gender and, if so, what is influencing such a gap.

The rest of this paper is organized as follows: Section 2 provides the background of the stimuli used in our study and related works. Section 3 describes the motivation for the study. Section 4 outlines our methodology, Section 5 summarizes the research findings, and Section 6 analyzes the results. Finally, Section 7 concludes the paper, and Section 8 provides suggestions for future work.

2. Materials and Methods

This section introduces Scratch, a block-based programming language, Dr. Scratch, which is an evaluator commonly used to analyze the CT score in projects and previous studies.

2.1. Scratch

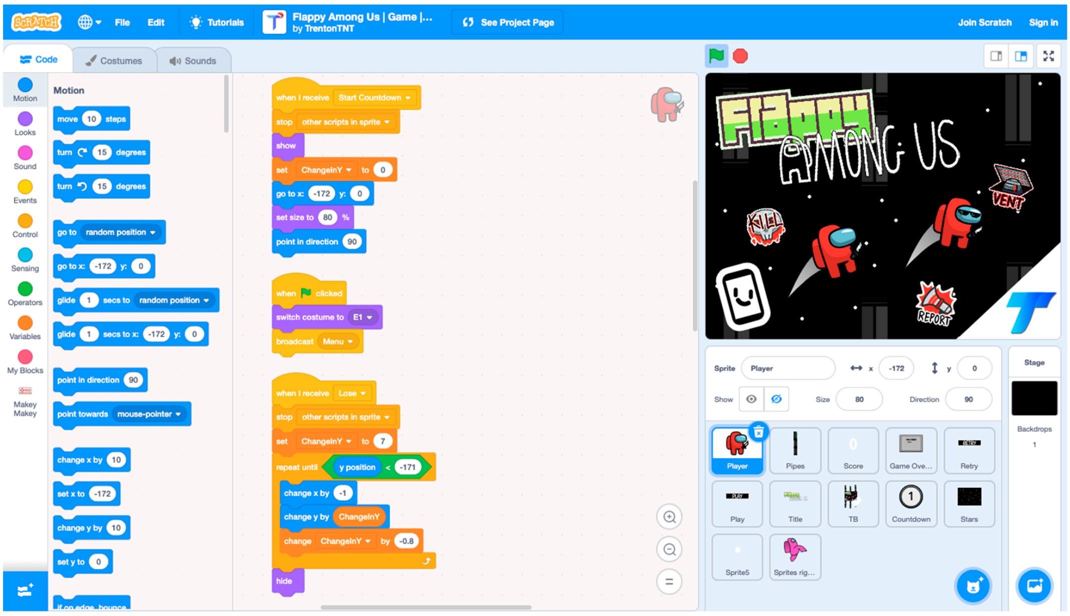
Scratch is a block-based visual programming language developed and maintained by the Lifelong Kindergarten group at the MIT Media Lab [9]. It is a popular educational tool intended for children aged 8–16 to promote CT skills, problem-solving skills, and equity in computing. Scratch users create and share interactive projects on the platform using a simple visual interface, which can be classified into five main types: animation, games, simulation, music, art, and stories [10]. This study analyzes Scratch projects to investigate the differences in coding trends by gender among younger programmers.

2.2. Dr. Scratch

Dr. Scratch is a web application that automatically analyzes projects coded on the Scratch platform [11]. This tool improves Scratch users' CT skills by providing detailed scores and feedback. Although it was initially developed to support effective assessments of students' projects and basic computer programming education in schools, it can also motivate students to improve their programming skills. Table 1 shows the seven criteria that Dr. Scratch uses to evaluate Scratch projects: flow control, data representation, abstraction, user interactivity, synchronization, parallelism, and logic. Owing to each criterion being evaluated on a three-point scale, the maximum score is 21. As examples of Scratch projects, Figures 1 and 2 show a game project and an art project, respectively. The former has a total Dr. Scratch score of 20, while the latter shows a total score of 2. The game project demonstrates more complexity in terms of the number of blocks used and their combinations.

Table 1. Dr. Scratch's evaluation criteria.

Category	Criterion
Flow Control	Whether the behavior of characters is controlled through repeated blocks.
Data Representation	How the position, facing direction, size of characters, etc., are set.
Abstraction	Whether the program is broken into parts that are easier to understand.
User Interactivity	Whether the project performs actions that invoke new situations.
Synchronization	How characters are organized to make movements happen in the intended order.
Parallelism	Whether the project can simultaneously run several things.
Logic	Whether the project behaves differently depending on the situation.

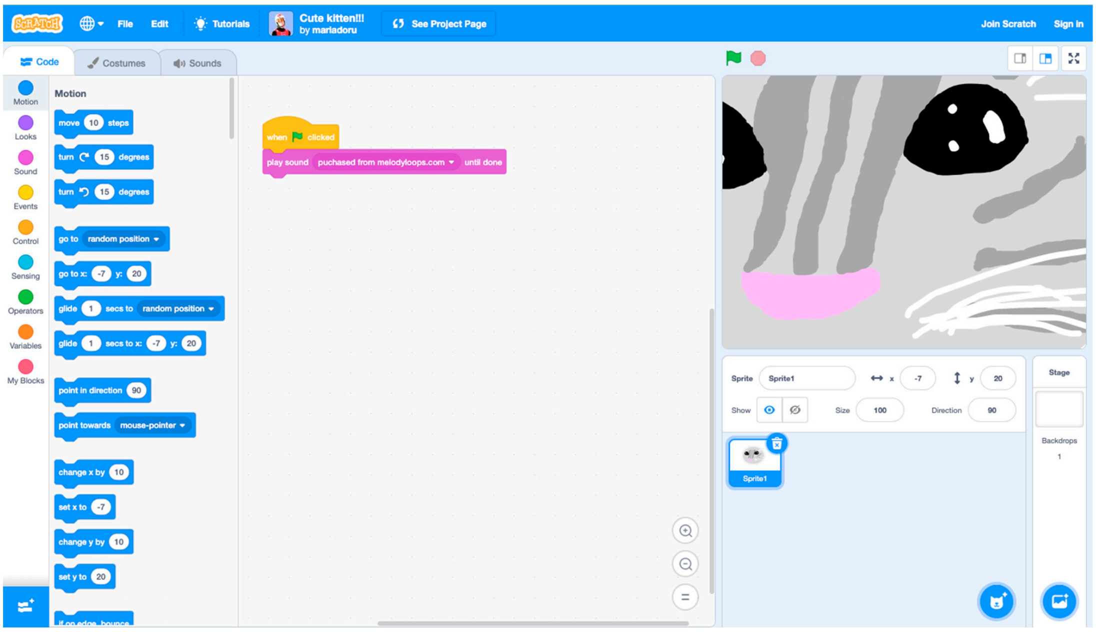


```
when I receive [Start Countdown]
stop [other scripts in sprite]
show
set [ChangeInY] to (0)
go to x: (-172) y: (0)
set size to (80) %
point in direction (90)

when 🏁 clicked
switch costume to [E1]
broadcast [Menu]

when I receive [Lose]
stop [other scripts in sprite]
set [ChangeInY] to (7)
repeat until <(y position) < (-171)>
    change x by (-1)
    change y by (ChangeInY)
    change [ChangeInY] by (-0.8)
hide
```

Figure 1. An example of a game Scratch project with a score of 20 on Dr. Scratch.



```
when 🏁 clicked
play sound [puchased from melodyloops.com] until done
```

Figure 2. An example of an art Scratch project with a score of 2 on Dr. Scratch.

2.3. Preliminary Study

A previous study compared the gender and CT score produced by Dr. Scratch for 60 Scratch users (30 males and 30 females) using open Scratch projects acquired from the Scratch website [12]. The gender information was extracted from self-reported data in each user's profile. Table 2 summarizes the total and category scores by gender. The CT score and gender showed a statistically significant link. Analysis of each project revealed that female users' projects lacked the Dr. Scratch-defined aspect of synchronization, which may be responsible for the gender score disparity.

Table 2. Total score by gender.

Gender	Number of Projects	Mean	Median	Mean Rank	Sum of Rank
Male	30	15.00	15	25.63	769
Female	30	16.50	18	35.37	1061

2.4. Related Works

Various studies have analyzed the CT skills of children using Scratch projects. Lawanto compared the CT skills of 360 seventh- and eighth-grade students engaged in a Scratch programming environment using Dr. Scratch. Students tended to miss the abstraction and data representation elements, which are two essential abilities in CT to simplify and identify necessary tasks [13]. This shortcoming was attributed to age because the users were too young to fully comprehend concepts such as abstraction and data representation.

Oluk and Korkmaz analyzed Scratch code by gender using Scratch projects prepared during a workshop involving 31 fifth-graders who developed Scratch projects within the framework of information technologies and software classes. The workshop began by teaching basic programming using Scratch for six weeks. Subsequently, the students' programming skills were assessed via Dr. Scratch. Although the students' gender and the obtained scores were not found to be correlated with statistical significance, the students' CT and Scratch skills were significantly related [14]. Other studies have analyzed the CT skills of beginner programmers developed in the Scratch environment owing to Scratch projects and CT skills being correlated [15–18].

Aivaloglou et al. presented an open database of 250,163 scraped Scratch projects to facilitate quantitative research in source code analysis and computing education [19]. Their data size was statistically sound; however, it had a limitation when creating a project portfolio since the only demographic information in the dataset was the username. The author noted that richer user data (e.g., gender and age) are essential to extend the research. The author also explained that demographic information could not be extracted from the current Scratch web interface because it had been omitted from the user profile. Furthermore, other studies have analyzed the massive datasets available on the Scratch website, although they were unable to collect adequate user information [20–22].

Moreno-Leon et al., Dr. Scratch's developers, showed that different types of projects could be used to develop distinct CT dimensions by performing a K-means cluster analysis on 500 projects randomly downloaded from the Scratch website [23]. That is, Dr. Scratch has shown discriminant validity to distinguish between different Scratch project types.

Through a three-day workshop, Funke and Geldreich explored the gender disparity in primary school students' Scratch programs [24]. Boys and girls employed different kinds of blocks to create their programs. For instance, boys used motion-related blocks twice as often as girls, whereas girls used visual-related blocks twice as often as boys. In addition, they discovered that the kinds of initiatives produced within the gender groups varied.

To identify the gender differences and similarities in Scratch programs, Graßl et al. conducted a topic analysis employing unsupervised machine learning in the programs [25]. They examined 317 Scratch projects and duplicated Funke and Geldreich's [24] basic programming course. Girls favored tales and animations, which led to more straightforward control systems. Meanwhile, boys developed games, leading to increasingly complicated programs.

Espino and González acknowledge the existence of gender gaps in computer processes between boys and girls but highlight that everyone is capable of developing CT skills [26]. Notably, only a few countries have committed to integrating CT into their educational curricula. They further claim that few approaches have been established expressly for teaching CT in early education, and none include a gender perspective. These findings highlight the vital need for developing a tutorial for educators, particularly those in early childhood and primary schools, to understand gender behaviors.

3. Motivation

Understanding the gender dynamics of early CS education might elucidate the origin of the sizable gender gap in this discipline. Previous research has examined gender trends in Scratch projects. For instance, Funke and Geldreich [24] and Graßl et al. [25] have investigated users' stereotypical gender characteristics; however, they did not consider the connection between the traditional gender traits in Scratch projects and crucial coding abilities such as CT.

In addition, previous research faced challenges due to limited sample sizes and a lack of user information. Although the user portfolio of Oluk and Korkmaz [14] has richer information than that of Aivaloglou et al. [19], the sample size was still limited. Hence, the outcome may be specific to the area where the research was performed, and so the existing dataset inhibits drawing a broader conclusion. Furthermore, a method such as coursework spread over time may impact the study length. Thus, a more effective approach is required. Further, the sample size in Aivaloglou et al. [19] seems promising, but the number of nominal variables is insufficient for hypothetical testing.

Due to the aforementioned limitations observed in previous studies, the relationship between stereotypical gender characteristics and CT remains unclear. This issue limits equal learning and the opportunity to diversify CS education in terms of gender. Promoting gender diversity in early education can increase exposure to traditionally marginalized groups in CS and eventually encourage them to pursue a career in the field. This leads to fostering the development of inclusive and user-friendly technological solutions in the field of CS.

Hence, our study includes a dependent variable—CT score—and two independent variables—binary gender roles and project type. This study contributes to the CS education field by duplicating an earlier preliminary study [12], with almost double the number of projects, and introducing a new independent variable, "project type," for additional research. Moreover, since both project type and CT score are introduced as variables, we aim to overcome the limitations of the studies by Funke and Geldreich [24] and Graßl et al. [25] and examine the connection between traditional gender traits in Scratch projects and CT. Furthermore, considering the dilemma of user information and sample size mentioned by Oluk and Korkmaz [14] and Meerbaum-Salant et al. [15], publicly accessible Scratch projects are assessed by using self-reported gender information to increase the sample size and collect user information. By retrieving such gender information, we attempt to combat the limitation of the lack of gender perspective in CT education mentioned by Espino et al. [26].

4. Methodology

Our work differs from previous studies because self-reported gender information is incorporated. Due to the limited user information available on the Scratch website, previous studies, which analyzed massive Scratch projects extracted from the Scratch website, had difficulty investigating the relationships between several variables. Our study attempted to tackle this problem. In addition, Aivaloglou et al. [19] highlighted that the restricted number of independent variables might result in an incorrect causal relationship between gender and CT score. To overcome this, this study introduced a new independent variable called project type in addition to the existing two variables, that is, gender (independent variable) and CT score (dependent variable). It examined the main influence of the score obtained from Dr. Scratch. Figure 3 visualizes the methodological design of the study.

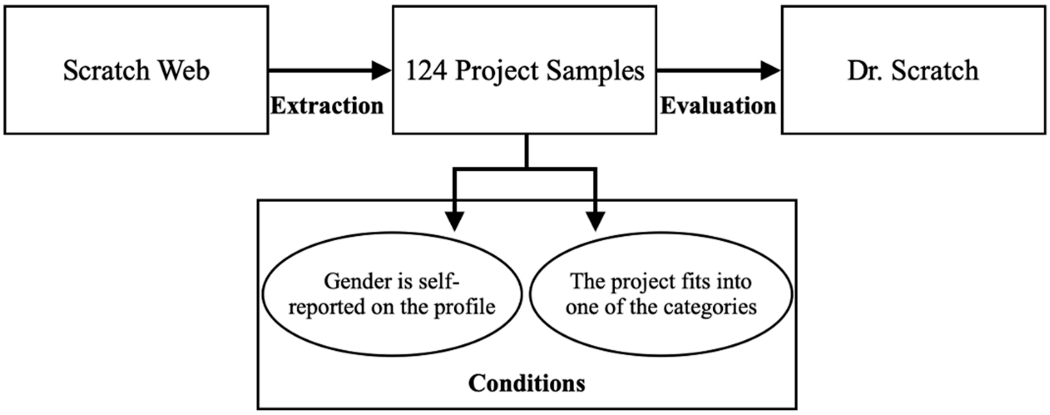

Scratch Web
Extraction
124 Project Samples
Evaluation
Dr. Scratch
Gender is self-reported on the profile
The project fits into one of the categories
Conditions

Figure 3. Methodological design.

4.1. Dataset and Materials

The analysis involved the open projects of 124 Scratch users (62 males, 62 females). The total sample was approximately 0.000113% of the open projects shared on the Scratch website [27]. Despite the very small sample size, we reached our target by doubling the size of the previous study. Users meeting the following conditions were chosen randomly: (1) the user declared their gender in the biography section of their page, and (2) the project could be categorized as one of the main project types [10]. Dr. Scratch analyzed and assessed the CT score.

4.2. Procedure

The procedure employed five steps. All the steps involving data collection were performed manually.

1. Select a random Scratch project from the Scratch Explore page in the "All" section, where all types of projects are posted by users [28].
2. Check the project creator's profile to determine if they have declared their gender information in their biography.
3. If gender information is present, classify the project based on Scratch's "Main Project Types" as animation, game, simulation, music, art, or story [10].
4. Paste the project's URL on the Dr. Scratch platform to assess the project.
5. Summarize the score details on an Excel sheet.

5. Results

Different combinations of variables were employed for investigating relationships: gender, project type, and CT score. As the dataset did not show a normal distribution, statistical testing that did not require a normal distribution for each variable pair was employed. In terms of project classifications, simulation, music, and story projects appeared less frequently than the other types. Considering the sample size, the scattered data for these categories were insufficient for analysis in independent categories; therefore, they were grouped as "other." Hence, four project types were considered in the analysis: game, animation, art, and other. The significance level, α, was set to 0.05.

5.1. Summary

Table 3 summarizes the mean, standard deviation, and the number of projects by gender. Figure 4 visualizes the data in Table 3 as a bar graph. Art projects had the lowest mean score, while game projects scored the highest. These trends were consistent for male and female users. In addition, the difference in the average total score of males by project types of art and games was more significant than that of females.

Table 3. Categorical score by gender.

Gender	Project Type	Mean	Standard Deviation	Number of Projects
Male	Animation	14.20	4.54	10
	Art	8.50	6.02	6
	Game	17.63	3.35	35
	Other	14.18	4.17	11
	Total	15.58	4.81	62
Female	Animation	14.50	4.38	14
	Art	11.15	3.56	13
	Game	16.52	2.50	23
	Other	12.42	3.06	12
	Total	14.15	3.90	62

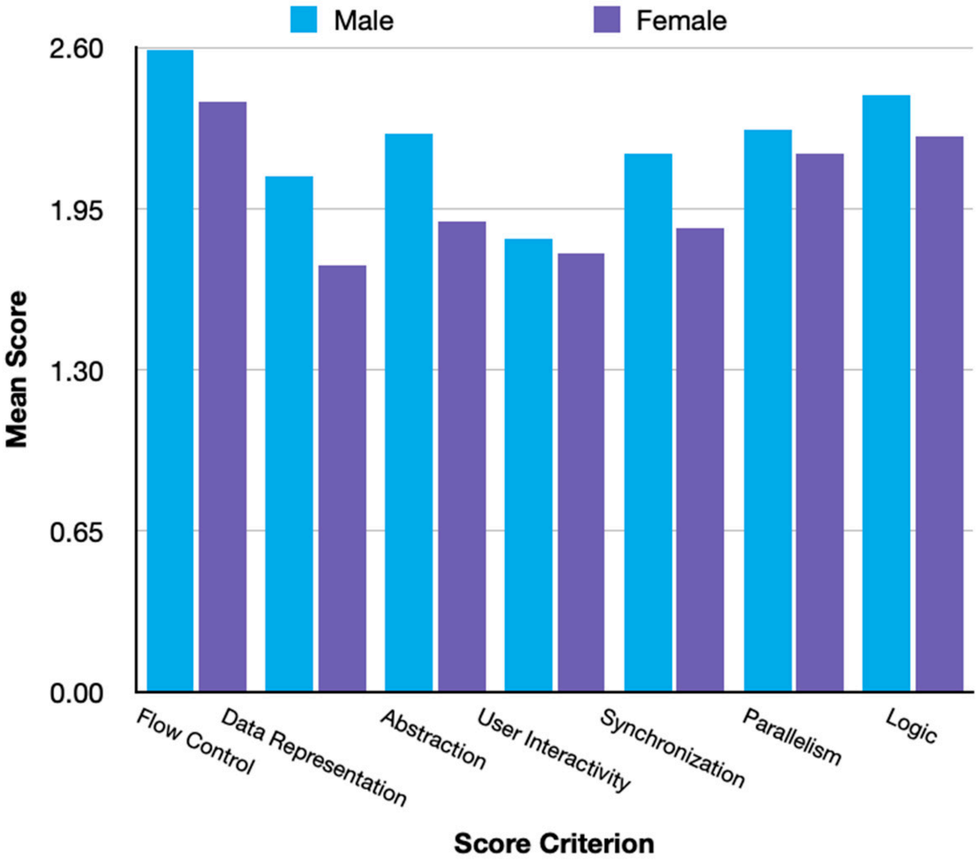


Figure 4. Mean score by project type.

Figure 5 plots the mean score for each criterion as a bar graph. Male and female users showed similar trends. Notably, both groups showed comparable scores for four criteria: flow control, user interactivity, parallelism, and logic. In contrast, significant differences were observed between males and females in data representation, abstraction, and synchronization. Male users scored the highest in flow control and the lowest in user interactivity. Meanwhile, female users scored the highest in flow control and the lowest in data representation.

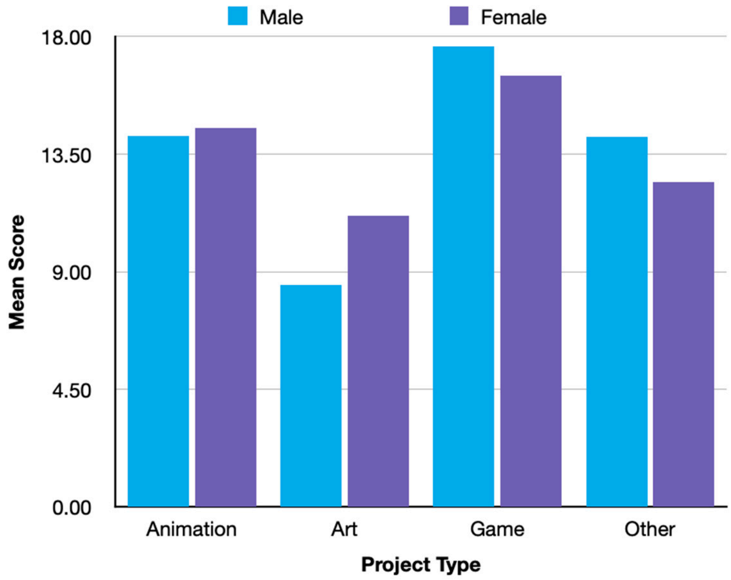


Figure 5. Mean score by score criteria.

5.2. Gender vs. Project Type

We performed a chi-squared test to investigate the relationship between gender and project type. The hypotheses for the test were as follows:

- Null Hypothesis: Male and female groups tend to select the same project type.
- Alternative Hypothesis: Male and female groups tend to select different project types.

The test yielded a p-value of 0.123 (chi-squared: 5.77, degrees of freedom: 3), indicating that a statistically significant difference was not observed; the null hypothesis cannot be rejected.

5.3. Total Score vs. Project Type

Figure 6 shows a significant gap between the scores in art and game projects. Art projects were broadly distributed, with a relatively low median of 11, whereas game projects were biased toward the top score, with a median of 18.

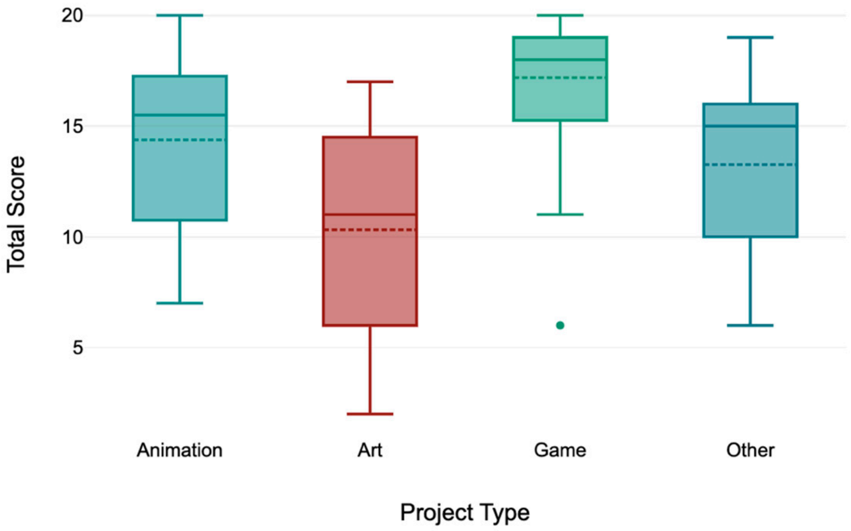


Figure 6. Total Score by Project Type.

We performed the Kruskal–Wallis test to determine whether a statistically significant difference was present between the dependent variable, total score, and the independent variable—project type. The hypotheses were as follows:

- Null Hypothesis: All project types perform similarly as regards the total score.
- Alternative Hypothesis: There are differences in performance between project type and total score.

The test yielded a *p*-value of <0.001 (chi-squared: 40.87, degrees of freedom: 3), indicating a statistically significant difference. Consequently, the null hypothesis was rejected, and the alternative hypothesis was accepted.

5.4. CT Score vs. Gender

We performed the Mann–Whitney U test to observe the relationship between CT score and gender.

5.4.1. Total CT Score

The total score was used as the dependent variable. The hypotheses for the test were as follows:

- Null Hypothesis: Male and female groups perform similarly with respect to the total CT score.
- Alternative Hypothesis: Male and female groups perform differently with respect to the total CT score.

The test yielded a *p*-value of 0.007 (Mann–Whitney U: 1381.5, z-value: −2.71), indicating a statistically significant difference. Thus, the null hypothesis was rejected, and the alternative hypothesis was accepted.

5.4.2. Criterion-Specific CT Score

Next, we analyzed the difference in performance between the male and female groups by the score criteria in Dr. Scratch. Table 4 shows the values obtained from the Mann–Whitney U test. We introduced the following hypotheses (note that the phrase "score criterion" is replaced by the name of the specific score criterion):

- Null Hypothesis: Male and female groups perform similarly with respect to the score criterion.
- Alternative Hypothesis: Performance differs between male and female groups regarding the score criterion.

Table 4. Mann–Whitney U test: gender vs. score.

Score Criterion	Mann–Whitney U	Z-Value	*p*-Value
Flow Control	1550.5	−2.11	**0.035**
Data Representation	1405.5	−2.78	**0.005**
Abstraction	1546	−2.06	**0.039**
User Interactivity	1812	−0.76	0.447
Synchronization	1434	−2.6	**0.009**
Parallelism	1794.5	−0.73	0.465
Logic	1747.5	−1.07	0.285

The *p*-values bolded in the table showed a statistically significant difference.

Table 4 showed a statistically significant difference, indicating that the score differences by gender were significant for the four criteria highlighted in bold. Consequently, the null hypothesis was rejected for flow control, data representation, abstraction, and synchronization.

5.4.3. Project Type-Specific CT Score

We classified the data by project type and then analyzed the relationships of the project types with gender. Table 5 shows the values obtained from the Mann–Whitney U test. We introduced the following hypotheses (note that the phrase "project type" is replaced with the name of the specific project type):

- Null Hypothesis: Male and female groups perform similarly with respect to project type.
- Alternative Hypothesis: Performance differs between the male and female groups regarding project type.

Table 5. Mann–Whitney U test: gender vs. project type-specific score.

Project Type	Mann–Whitney U	Z-Value	*p*-Value
Animation	68.5	−0.09	0.928
Art	29.5	−0.80	0.401
Game	249.5	−2.48	0.013
Other	43	−1.4	0.153

Table 5 showed a statistically significant difference for game projects. Therefore, the null hypothesis was rejected for the game project, indicating that only the score for game projects differed by gender.

5.5. Evaluation of the Four Criteria

As the flow control, data representation, abstraction, and synchronization criteria showed significant differences, we attempted to identify the relationships between them and the project type (Figure 7). Only the game projects consistently scored high for all criteria. In contrast, art project scores were significantly lower than the other project types for all criteria.

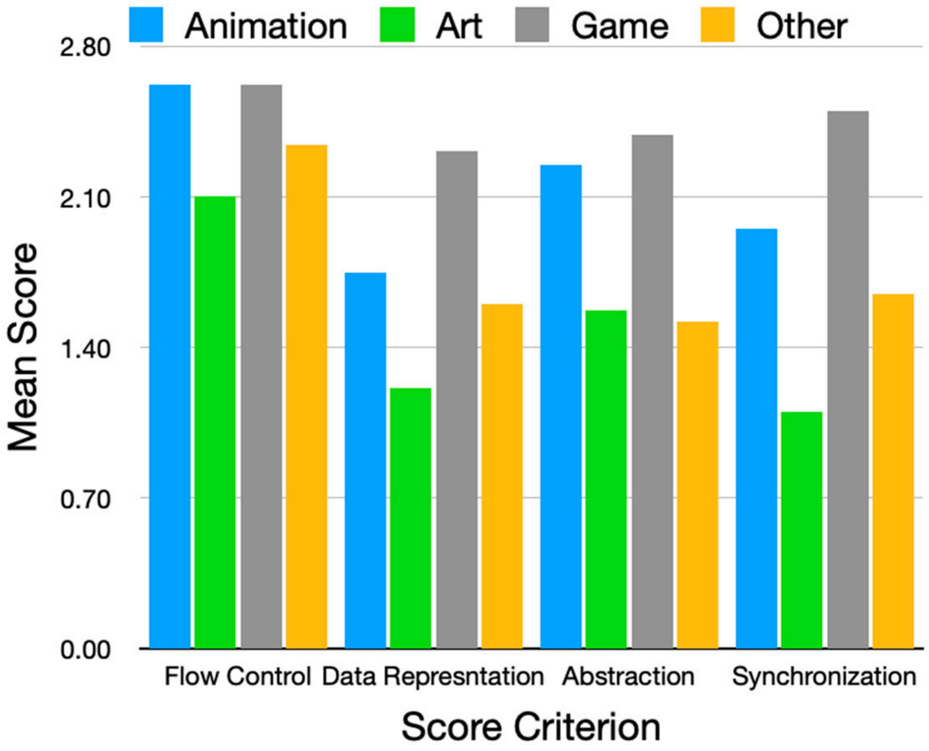


Figure 7. Mean score by the four score criteria.

6. Discussion

6.1. Analysis of the Results

Although the testing performed in Section 5.2 did not show statistical significance, enlarging the sample size may affect the results because the *p*-value was close to 0.05. In addition, Table 4 shows a relatively biased selection of project types by gender. Of the

62 projects by male users, only 6 were art projects, while 35 were game projects. Such extreme differences in the selection were not observed in the female group.

RQ1. Does gender influence the type of Scratch project selected?

The results are inconclusive about the role of gender in the selection of the type of project. However, further investigation is necessary because the *p*-value is close to the significance level, suggesting that increasing the sample size may reveal a difference in project type by gender.

RQ2. Do certain project types demonstrate higher CT scores?

Our findings support the notion that some project types scored higher than others. For example, game projects scored high, while art projects scored low for almost all evaluation criteria. The scores revealed a gender difference. This gap was likely caused by the criteria, which revealed statistically significant differences. However, Figure 7 suggests that the underlying reason for the significant differences in the four criteria is that they tended to generate relatively high scores in game projects, which were selected more frequently by the male group. In addition, the score difference by gender for project types was insignificant, except for game projects. Given that game projects tended to have higher scores than other project types, the difference in sample size may be disadvantageous for the female group because it had 23 game projects while the male group had 35. For instance, the female group scored the lowest in data representation, which had poor performance, especially in art projects. The female group selected art projects more frequently than the male group, and the characteristics of art projects likely produced a low score rather than an inherent inability of females to represent data correctly. Regardless, the project category contributed to the score difference.

RQ3. Is there a gender gap in CT skills?

Although there was a difference in performance by gender, this is likely due to the compatibility of certain project types with Dr. Scratch rather than any skill differences between genders. Project types preferred by males generally require more programming constructs, which are reflected in the CT score.

6.2. Threats to Validity

6.2.1. Internal Validity

This study analyzed the data through different types of hypothetical testing and concluded that the differences in scores might be due to the project type. As our method can only reject or accept the null hypothesis between variables, further analysis is needed to determine the causal inference distinctly.

Our method used Dr. Scratch as the measuring tool to evaluate CT skills. While Dr. Scratch evaluates the criteria indicated in Section 2.2, it does not measure other vital aspects of CT, such as dead code, attribute initialization, sprite naming, or repeated code. Hence, the CT scores do not entirely represent a user's CT skills.

This study only considered projects with self-reported user information. This limited selection method may not directly represent gender trends because a limited number of users declare their gender on Scratch. Moreover, it is unclear whether both genders are equally likely to self-report their gender. This is another reason why relying on self-reported gender is a limitation.

6.2.2. External Validity

The data were collected manually because user information such as gender and age is not available on the current Scratch web interface. This resulted in a significantly limited number of samples. With a sample size equivalent to only 0.000113% of the total number of Scratch projects on the official website, it remains difficult to conclude that the results are, as of now, entirely representative of the larger population. Moreover, manual data collection may have introduced human errors during the data collection phase.

We acknowledge the lack of independent variables in this study. Demographic information, such as age, location, and personal interest, may affect the results. In addition, if a project is created in a classroom setting, there is a possibility that students are assigned a particular type of project by their teacher and do not have the option to choose on their own. Since this study did not consider such external biases when evaluating, not all selected project types correctly represent each user's preference.

Due to the limited sample size of simulation, music, and story projects, they were all classified into the "other" category. However, these project types may have had different CT characteristics. This study failed to identify performance differences within genders as the three types were not evaluated in independent categories.

7. Conclusions

This study evaluated the effect of binary gender preferences on CT skills via Scratch programs. As Dr. Scratch is compatible with specific project types, such as game projects, a given project type may have a larger impact on the score than gender. Consequently, it is difficult to conclude that the scores obtained through Dr. Scratch represent the CT skills of an individual. Although we could not support the hypothesis that gender plays a role in the selection of project types, increasing the sample size may yield a different result. With the current distribution of project types indicated in Table 3, males' preference for games is very high compared to that for other project types. In contrast, females' preferences are distributed more evenly across different categories. Based on the characteristics stated by Graßl et al. [25] in Section 2.4, we assume that these differences in selection between genders may have connections with stereotypical gender preferences. Our study had some limitations. The major limitation was a lack of sample size due to manual data collection, which may have also introduced human errors. Moreover, the lack of independent variables and Dr. Scratch's limitations with returning a score that represents CT holistically were also challenges of the study.

This research provides insight for both researchers and educators. Researchers conducting studies involving Dr. Scratch should classify projects carefully before analyzing the data. Failure to do so may yield an incorrect causal relationship between the CT score and the project creator's CT skill. Educators, such as teachers who implement Scratch in an educational environment, should design curricula such that students can improve their CT regardless of the project they choose. Instead of relying solely on CT scores, educators should attempt to manually evaluate aspects that are not covered by the current scoring criteria, such as the visual engagement of projects. Finally, STEM education content providers should create content appealing to both genders, for instance, by using examples from popular culture or everyday life. This can demonstrate how STEM concepts are relevant for all students, regardless of gender. These approaches would provide a less biased evaluation of CT between gender groups.

8. Future Work

To address the limitations of Dr. Scratch, which fails to detail some vital aspects of CT, future research can expand the evaluation criteria to include missing aspects of the current evaluation system. For instance, to measure the dead code, a metric to evaluate the percentage of code blocks that are not used or executed in the project could be created. Tools such as the Scratch analysis tool (SAT) are new CT evaluation tools that attempt to overcome the limitations of Dr. Scratch [29]. Utilizing such new tools as a substitute for, or in addition to, Dr. Scratch may provide a more holistic evaluation of CT skills.

For maximizing the number of project types with self-reported gender information, future research could automate data collection by web scraping and implementing text data mining techniques. Thus, unstructured gender information on the user profile can be transformed into texts to identify patterns and classify projects based on gender.

Additionally, to account for the limitation where not all project types selected can represent each user directly due to external biases, future research could explore the project descriptions or the comments section. Some users may provide information about the project's origin in the description section, which may mention whether the project was created for a specific class or assignment. However, projects including such information may be in the minority, as users frequently omit project descriptions on the Scratch website. Therefore, researchers should consider conducting this study in a classroom setting, where they design the study to allow the students to choose the type of project. This offline study design can potentially overcome the lack of user information.

Moreover, it would be interesting to analyze and compare CT scores for different project types by a specific user. Another potential direction is to develop a Scratch assessment tool that adds a new feature to classify projects by type before providing the CT score.

We hope to expand the scope of this study such that educators can design CS educational tools with which students can enhance their CT skills without being limited by external factors such as gender preferences. Eventually, we aspire to develop a curriculum that strengthens CT skills even through art and other visual-based project types.

Author Contributions: Conceptualization, methodology, formal analysis, investigation, resources, data curation, writing—original draft preparation, R.N.; writing—review and editing, R.N., D.S. and H.W.; supervision, D.S., H.W. and Y.F. project administration; all authors. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data are available upon request.

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

References

1. Soomro, T.R. Stem education. In Proceedings of the 2019 8th International Conference on Educational and Information Technology, ICEIT, Cambridge, UK, 2–4 March 2019; pp. 157–160.
2. Gender Equity and Education. Available online: https://www.unesco.org/en/gender-equality/education (accessed on 19 January 2023).
3. 40 Telling Women in Technology Statistics [2023]: Computer Science Gender Ratio. Available online: https://www.zippia.com/advice/women-in-technology-statistics/ (accessed on 19 January 2023).
4. Rich, P.J.; Browning, S.F.; Perkins, M.; Shoop, T.; Yoshikawa, E.; Belikov, O.M. Coding in K-8: International trends in teaching elementary/primary computing. *TechTrends* **2018**, *63*, 311–329. [CrossRef]
5. Fagerlund, J.; Häkkinen, P.; Vesisenaho, M.; Viiri, J. Computational thinking in programming with scratch in primary schools: A systematic review. *Comput. Appl. Eng. Educ.* **2020**, *29*, 12–28. [CrossRef]
6. Wing, J.M. Computational thinking. *Commun. ACM* **2006**, *49*, 33–35. [CrossRef]
7. Li, Y.; Schoenfeld, A.H.; diSessa, A.A.; Graesser, A.C.; Benson, L.C.; English, L.D.; Duschl, R.A. Computational thinking is more about thinking than computing. *J. STEM Educ. Res.* **2020**, *3*, 1–18. [CrossRef] [PubMed]
8. Pat, P. Computational thinking: A problem-solving tool for every classroom. *CSTA* **2009**, 12–16.
9. Resnick, M.; Silverman, B.; Kafai, Y.; Maloney, J.; Monroy-Hernández, A.; Rusk, N.; Eastmond, E.; Brennan, K.; Millner, A.; Rosenbaum, E.; et al. Scratch: Programming for All. *Commun. ACM* **2009**, *52*, 60–67. [CrossRef]
10. Scratch Project Types. Available online: https://en.scratch-wiki.info/wiki/Project_Types (accessed on 19 January 2023).
11. Moreno-León, J.; Robles, G. Dr. Scratch. In Proceedings of the Workshop in Primary and Secondary Computing Education, London, UK, 9–11 November 2015; pp. 1–23.
12. Niousha, R.; Saito, D.; Washizaki, H.; Fuakzawa, Y. Scratch Project Analysis: Relationship Between Gender and Computational Thinking Skill. *IEEE TALE*, 2023; *accepted*.
13. Lawanto, K.N. Exploring Trends in Middle School Students' Computational Thinking in the Online Scratch Community: A Pilot Study. Master's Thesis, Utah State University, Logan, UT, USA, 2016.
14. Oluk, A.; Korkmaz, Ö. Comparing students' scratch skills with their computational thinking skills in terms of different variables. *Online Submiss.* **2016**, *8*, 1–7. [CrossRef]
15. Meerbaum-Salant, O.; Armoni, M.; Ben-Ari, M. Learning Computer Science Concepts with Scratch. In Proceedings of the Sixth International Workshop on Computing Education Research, Aarhus, Denmark, 9–10 August 2010.
16. Liao, C.-H.; Hsu, H.-J.; Wu, P.-C. Integrating Computational Thinking in Math Courses for 3rd and 4th Grade Students with Learning Disabilities via Scratch. In Proceedings of the 51st ACM Technical Symposium on Computer Science Education, Portland, OR, USA, 11–14 March 2020.

17. Ford, A.; Hainey, T.; Connolly, T. Evaluation of Computer Games Developed by Primary School Children to Gauge understanding of Programming Concepts. In Proceedings of the European Conference on Games Based Learning, Cork, Ireland, 4–5 October 2012.
18. Heintz, F.; Mannila, L. Computational thinking for All. *ACM Inroads* **2018**, *9*, 65–67. [CrossRef]
19. Aivaloglou, E.; Hermans, F.; Moreno-Leon, J.; Robles, G. A Dataset of Scratch Programs: Scraped, Shaped and Scored. In Proceedings of the 2017 IEEE/ACM 14th International Conference on Mining Software Repositories (MSR), Buenos Aires, Argentina, 20–21 May 2017.
20. Fields, D.A.; Giang, M.; Kafai, Y. Programming in the Wild. In Proceedings of the 9th Workshop in Primary and Secondary Computing Education, Berlin, Germany, 5–7 November 2014.
21. Dasgupta, S.; Hale, W.; Monroy-Hernández, A.; Hill, B.M. Remixing as a Pathway to Computational Thinking. In Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing, Minneapolis, MN, USA, 14–18 October 2023; pp. 1438–1449.
22. Yang, S.; Domeniconi, C.; Revelle, M.; Sweeney, M.; Gelman, B.U.; Beckley, C.; Johri, A. Uncovering Trajectories of Informal Learning in Large Online Communities of Creators. In Proceedings of the Second (2015) ACM Conference on Learning @ Scale, Vancouver, BC, Canada, 14–18 March 2015; pp. 131–140.
23. Moreno-Leon, J.; Robles, G.; Roman-Gonzalez, M. Can we Measure Computational Thinking with Tools? In Proceedings of the Seminar Series on Advanced Techniques and Tools for Software Evolution SATToSE, Madrid, Spain, 7–9 June 2017.
24. Funke, A.; Geldreich, K. Gender Differences in Scratch Programs of Primary School Children. In Proceedings of the 12th Workshop on Primary and Secondary Computing Education, Nijmegen, The Netherlands, 8–10 November 2017; pp. 1–9.
25. Graßl, I.; Geldreich, K.; Fraser, G. Data-Driven Analysis of Gender Differences and Similarities in Scratch Programs. In Proceedings of the 16th Workshop in Primary and Secondary Computing Education, Virtual Event, Germany, 18–20 October 2021; pp. 1–10.
26. Espino, E.; González, C. Gender and Computational Thinking: Review of the Literature and Applications. In Proceedings of the XVII International Conference on Human Computer Interaction, Salamanca, Spain, 13–16 September 2016; pp. 1–2.
27. Community Statistics at a Glance. Available online: https://scratch.mit.edu/statistics/ (accessed on 19 January 2023).
28. Scratch Explore. Available online: https://scratch.mit.edu/explore/projects/all (accessed on 19 January 2023).
29. Chang, Z.; Sun, Y.; Wu, T.; Guizani, M. Scratch Analysis Tool (SAT): A Modern Scratch Project Analysis Tool based on ANTLR to Assess Computational Thinking Skills. In Proceedings of the 2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC), Limassol, Cyprus, 25–29 June 2018.

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.