IIAI Open Conference Publication Series
IIAI Letters on Institutional Research
Vol.004, LIR300
DOI: https://doi.org/10.52731/lir.v004.300

Programming Education Methods for Elementary School Students and Their Relation with Personal Preferences

Emi Ichimura *, Tomonari Kamba *

Abstract

This study investigated the differential impact of cooperative and competitive instructional strategies in the programming education for elementary students using the visual programming language, Scratch. The methodology involved conducting 40-minute sessions within two distinct educational settings to explore how students' preferences for specific tastes, colors, and school subjects influenced their learning outcomes. These preferences were selected from readily accessible elements that could be acquired rapidly, thereby serving as indicators to facilitate a simplified assessment of students' personality traits. The efficacy of the instructional sessions was gauged by evaluating task achievement and ingenuity, which were further linked to personality traits extrapolated from student preferences. The results demonstrated that a competitive setting notably enhanced both achievement and ingenuity. Remarkably, students who preferred competitive environments exhibited higher levels of achievement and ingenuity, whereas most participants predominantly perceived cooperative environments as more enjoyable. No significant relationships emerged between learning outcomes and other preferences, such as gender, favorite color, or chosen academic subjects. This study highlights the critical importance of customizing programming instruction to align it with the individual characteristics and preferences of students to optimize educational effectiveness.

Keywords: Programming education, Children's education, Educational method

1 Introduction

The exponential growth of information technology (IT) has ushered in a global consensus regarding the indispensability of programming skills [1][2]. To ensure the sustained viability of IT professionals, cultivating a multifaceted repertoire of cognitive abilities in youth is as imperative as acquiring proficiency in specific technologies. Although programming education, particularly within primary settings, remains in its nascent stages, it has emerged as engaging and efficacious pedagogical content, garnering significant attention [3]. Diverse learning modalities exist; however, short-term events, such as summer camps, can effectively galvanize student participation. This study comprehensively explored the variable effectiveness of various programming pedagogies, their interconnectedness, and their congruence with students' unique preferences and characteristics.

* Toyo University, INIAD, Tokyo, Japan

2 Related Research

2.1 Programming language for elementary school students

Programming languages in K-12 education (referring to 13 years of schooling before joining university) are typically classified into two main categories: visual and textual [4]. Textual programming is often viewed as more practical and primarily used in real-world software development [5]. Conversely, visual programming provides ease of access to beginners and can make learning experiences enjoyable [6]. Despite their distinct characteristics, early exposure to visual programming enhances the understanding of textual programming concepts [7]. Therefore, visual languages, such as Scratch, are frequently employed in elementary education [8][9].

2.2 Educational methods at school sites

In programming education, diverse educational methodologies have been proven to significantly enhance learners' confidence, along with providing numerous other benefits. Comparative studies of individual and pair programming indicate that, while skill acquisition levels are similar, pair programming markedly boosts learner confidence [10]. The educational environment is crucial for the development of academic skills. Although both cooperative and competitive settings enhance student focus and motivation [11][12], their effectiveness varies with subject. There is a report that gamified learning boosts motivation but not necessarily outcomes [13].

The previously discussed factors, especially the learning environment, require further investigation within the framework of programming education for elementary students. Schools encounter significant challenges in tailoring educational methods to individual learners owing to constraints, such as limited classroom time. Moreover, the selection of these pedagogical approaches must consider factors such as student gender and personality, which significantly impact their engagement with programming. Considering these individual differences, there is an urgent need to explore more effective teaching strategies that better meet the specific needs of students.

2.3 Relationship to personal attributes and preferences

Research in the science, technology, engineering, and mathematics disciplines has demonstrated variances in learning styles and motivation across genders [14][15]. Furthermore, studies indicate that these gender differences extend to coding skills [16], which subsequently impacts elementary school students' programming education. Specifically, the five-factor model (FFM), that categorizes personality into five distinct traits indicates that individuals with elevated levels of openness achieve better learning outcomes, particularly when engaged in paired programming [17]. Additionally, studies have been conducted to investigate how personality traits correlate with personal preferences, providing deeper insights into individual learning processes [18][19].

In this research situation, personality traits can be efficiently determined by examining gender and personality, as well as personal preferences. Existing personality assessments are time consuming. Measuring personality traits by preference can reduce the burden on respondents and help to rapidly identify appropriate educational methods. Therefore, this study employed a preference-based approach to effectively tailor program education strategies. Visual programming, widely used in elementary education, was selected for this study. This study evaluated the effectiveness of the implemented educational strategy by conducting experiments in elementary schools and analyzing the data obtained. In the following sections, the methodology used in this study is described in detail and the results are discussed as well.

3 Survey Methodology

3.1 Selection and Description of Participants

Two urban public elementary schools within the same region were selected as the experimental schools, because they had similar levels of student participation in club activities. A partnership was established with a nonprofit organization (NPO) that oversees the daily club activities. Through this partnership, the NPO provided data on the number of club activity participants at each school and assisted in supervising the students during the experiment. The study participants were children who voluntarily chose programming from among the various club activity options available at their respective schools, without the influence of NPO or researchers. Participants must be in the 4th, 5th, or 6th grade. These children had no exposure to programming before joining the club. They learned the basic skills required to manipulate character movements in Scratch in two sessions held before the start of the study. The experiment used Chromebooks loaned by the school in accordance with the Japanese educational policy, and the programming language used was Scratch.

3.2 Programming assignment

Owing to the time constraints of the school, a total of one hour was allotted for the entire project, with 40 minutes specifically designated for production. The task assigned to the students was to create movements for three characters: starfish, crabs, and clownfish. To facilitate the learning process, the students were instructed to follow a step-by-step approach.

- Step 1: Understand the concept of iteration (while statements) and program the starfish to move toward the mouse pointer. This step aimed to familiarize the students with the new concept by showing them how to operate it.
- Step 2: To introduce conditional branching (if statements), we incorporated the crab into the project and programmed it to speak when it touched a starfish.
- Step 3: Assess the understanding acquired in Steps 1 and 2. A clownfish was added to the project, and the concepts of while and if statements were used to program clownfish movements without guidance.

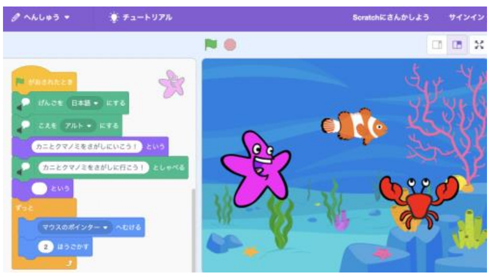


Figure 1: Screen shot example of the finished product

3.3 Implementation of educational activities

To analyze and compare educational methods, contrasting approaches were implemented in schools A and B:

- In School A (cooperative approach), students were paired with classmates, and each student was programmed to use his or her own device. Rather than sharing the roles of driver and navigator, the students worked closely with each other and served as consultants seeking advice and guidance as needed. Finally, each student submitted their final piece of work.
- School B (competitive approach): Students worked independently and competed for time to complete their projects. The first five students to complete the task were awarded additional stickers to create a sense of competition. Upon completion, all the participants received at least one sticker.

In both Schools A and B, students were provided with a printout detailing Steps 1–3 of their assignments. In each classroom, supported by two teachers and five to six non-profit volunteers, students followed the instructions and completed the task in 40 minutes.

3.4 Questionnaire

The questionnaires listed in Table 1 were distributed after the survey was administered. In particular, there were 27 and 22 participants from schools A and B, respectively; however, only 23 and 21 participants from schools A and B, respectively, returned completed questionnaires without any missing information in the required fields. The completed questionnaires were used in the subsequent analyses.

Table 1: Questionnaire items

	Question	Method of response
Q1	Did you enjoy programming [to a goal / with your pair]?	5 levels
Q2	What grade are you in?	Single (4th / 5th / 6th grade)
Q3	Gender.	Single (male / female /no answer)
Q4	Was the teacher's talk easy to understand?	5 levels
Q5	Do you want to do programming again?	5 levels
Q6	Favorite color	Single choice (red / orange / yellow / green / light blue / blue / purple)
Q7	Favorite food taste	Single choice (sweet / salty / sour / spicy)
Q8	Favorite school subject	3 choices (Japanese / Math / Science / Social Studies / English / Music / Arts and craft / Home Economics)
Q9	Free comment	Free answer (optional)

4 Result

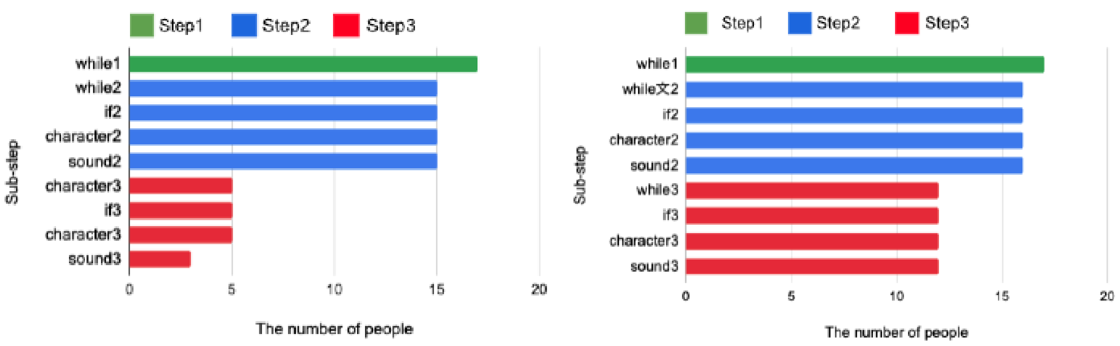
The results of Q2 and 3 are shown in Table 2.

Table 2: Respondent Attributes

School	Number of students	Boy	Girl	No answer	4th grade	5th grade	6th grade
A (Cooperative)	23	16	5	2	20	3	0
B (Competitive)	21	15	2	4	3	11	7

4.1 Educational methods and number of correct answers

An intriguing finding emerged during the administration of Step 3, in which no explicit instructions were provided. The rate of correct responses was significantly higher under competitive than under cooperative conditions. As illustrated in Figure 2, which disaggregates the tasks in Steps 1–3 into more granular sub-steps (such as changing the background color), a response was deemed correct if all sub-components were executed accurately. However, although the rates of correct responses in Steps 1 and 2 were similar, the rate of correct responses in Step 3 was significantly higher in the competitive than in the cooperative type. This disparity occurred despite the absence of significant differences in students' skill levels across both instructional modes. These results suggest that the instructional approach of competitive instruction may have contributed to the increase in the number of correct answers in terms of task completion within the time limit, which will be analyzed in more detail in subsequent sections.



(a) Cooperative (b) Competitive

Figure 2: Number of correct answers of the sub-steps in the programming assignment

4.2 Calculation of achievement and ingenuity

We introduced "achievement level" and "ingenuity level" as indicators to evaluate the results of this study. In the previous section, we focused only on the number of correct answers in the task. However, upon reviewing the results, there were instances in which the participants spontaneously attempted to change colors and movements without specific instructions. The ten items used to calculate the level of achievement and the four items used to calculate the level of ingenuity are shown in Tables 3 and 4.

Table 3: Achievement Score

Step	Item	Contents	Score	Partial sum	Total
Whole	A	Use of characters	4	4	13
Step 1	B	“while” use	1	1	
Step 2	C	“while” use	1	3	
	D	“if” use	1		
	E	Character display	0.5		
	F	Phonetic indication	0.5		
Step 3	G	“while” use	2	5	
	H	“if” use	2		
	I	Character display	0.5		
	J	Phonetic indication	0.5		

Table 4: Ingenuity Score

	Item	Content	Score	Partial sum	Total
Basic	K	Changed background	0.5	1	3
	L	Changed character	0.5		
Advanced	M	Use of 4 or more character	1	2	
	N	Use of camera motion	1		

The achievement level is structured to measure students’ understanding of programming concepts and their ability to execute them accurately. Rather than counting only the number of substeps completed, the scoring system focused on constructs that demonstrated a deeper understanding of programming. For example, constructs, such as "while" and "if" statements received high scores because of their complexity and their essential role in logical problem solving in programming. Conversely, items related to elementary tasks scored lower, indicating that the quality of programming understanding is more important than task completion. In particular, Step 3, where "while" and "if" statements are executed without explicit instructions, received a high score.

The ingenuity level was evaluated based on the execution of undirected tasks. For example, points are awarded for actions, such as replacing one character on the screen with another. Items K and L were associated with relatively straightforward alterations, whereas Items M and N involved more complex technical enhancements that surpassed the instructions provided.

5 Analysis

5.1 Comparison of Cooperative and Competitive Types

In the study, participants classified under the competitive type demonstrated statistically significant higher levels of 'achievement' and 'ingenuity' compared to those in the cooperative type. However, a greater proportion of children in the cooperative group reported finding programming activities to be fun. Because the competitive type yielded a higher number of correct responses, we conducted an analysis to discern differences in 'achievement' and 'ingenuity' between the two pedagogical approaches. The distributions of these metrics are presented in Table 5.

A comparative analysis revealed that School A, which employed a competitive pedagogical approach, outperformed School B, which utilized a cooperative approach in both achievement and ingenuity metrics. This disparity was substantiated by the Mann–Whitney U test, which confirmed statistically significant differences. Specifically, in the domain of achievement, the U-value was recorded at 150.5 with a p-value of 0.028, indicating significance at the level of 0.05. Similarly, for ingenuity, a U-value of 130 and p-value of 0.006 were noted, both of which substantiate the superior performance of the competitive type.

Table 5: Relationship between educational methods and achievement/ingenuity score

Item	School	Educational method	Number of students	Total	Mean rank
Achievement	A	Cooperative	23	44	26.5
	B	Competitive	21		18.2
Ingenuity	A	Cooperative	23	44	27.3
	B	Competitive	21		17.2

The subsequent analysis, which included detailed observations of the study and responses to the questionnaire, revealed the following insights:

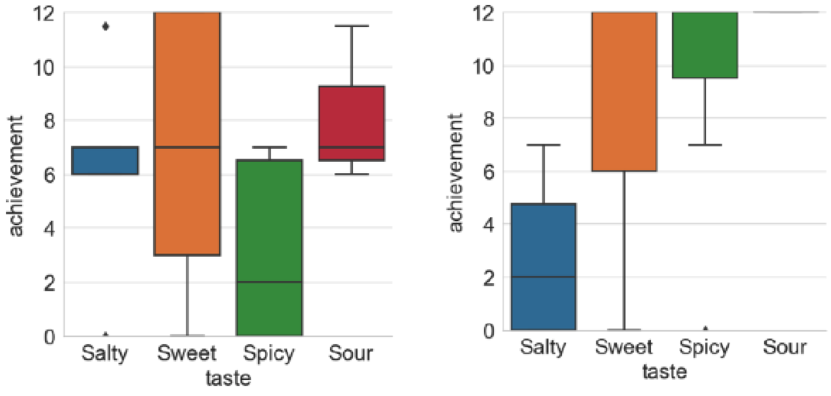
(1) The notably lower achievement rate in the cooperative type may stem from the fact that only approximately 21% of the students managed to independently create a third character in Step 3, which is significantly lower than the 57% observed in the competitive type. In terms of ingenuity, when faced with the task of having characters speak their lines, approximately 57% of the students (12 students) in the competitive type displayed the characters, as well as successfully implemented speech synthesis technology. Conversely, such innovative applications are rarely observed in the cooperative type.

(2) However, 15 out of 23 students (approximately 65%) in the cooperative type described their experience as 'very enjoyable' in the free-response section, compared to only approximately 30% of students in the competitive type, indicating a contrast where the cooperative type, although scoring lower, found the activity more enjoyable.

These observations indicate that although the competitive type was statistically significantly superior in terms of achievement and ingenuity, the cooperative type proved more effective in making children find programming enjoyable.

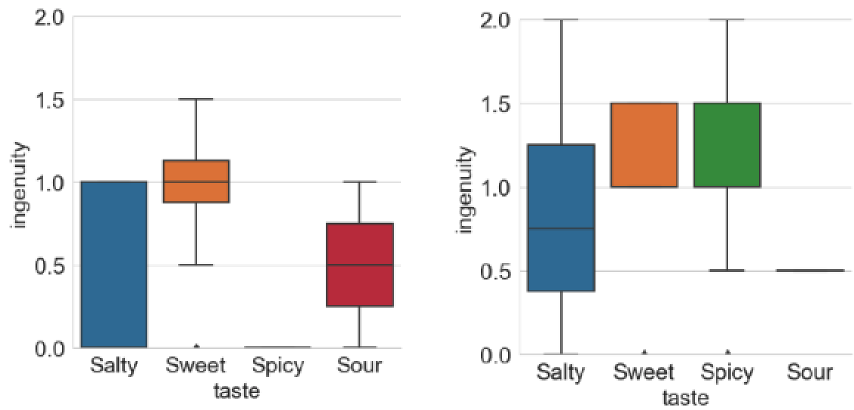
5.2 Relation with personal preferences

The preceding analysis demonstrated that participants categorized under the competitive model exhibited superior performance compared to those categorized under the cooperative type. Expanding upon these initial findings, the current section analyzes the potential correlations between individual characteristics and preferences. The skewed gender distribution, characterized by a disproportionately low number of female participants, precluded a comprehensive gender-based analysis. Moreover, the substantial heterogeneity in responses related to preferred colors and school subjects precluded the possibility of conducting a precise statistical analysis.



(a) Cooperative (b) Competitive

Figure 3: Achievement and favorite taste



(a) Cooperative (b) Competitive

Figure 4: Degree of ingenuity and favorite taste

The correlations between achievement, ingenuity, and preferred taste are illustrated in Figures 3 and 4. Among the 44 students surveyed, their preferences were as follows: 18 for spiciness, 13 for sweet, 9 for salty, and 4 for sour. The analysis primarily focuses on the two most popular tastes: spiciness and sweetness. The graphical representations indicate no significant differences in achievement or ingenuity among students with a preference for sweetness. Conversely, students favoring spicy tastes displayed notably higher scores in both achievement and ingenuity within the competitive type, as confirmed by the Mann-Whitney

U-test. Specifically, for spicy taste preferences, achievement registered a U-value of 7.5 with a p-value of 0.003 (<0.05), and ingenuity a U-value of 5.5 with a p-value of 0.0009 (<0.05). However, for students who preferred sweet tastes, the differences in achievement (U-value = 16.5, p-value = 0.641) and ingenuity (U-value = 17.5, p-value = 0.754) were not statistically significant, indicating no meaningful variance linked to the pedagogical approach.

Table 6: Achievement and ingenuity level according to favorite taste and its relation to educational methods

Favorite	Evaluation method	Group	Educational method	Number of students	Total	Mean rank
Spicy	Achievement	A	Cooperative	7	18	13.9
		B	Competitive	11		6.7
	Ingenuity	A	Cooperative	7	18	14.5
		B	Competitive	11		6.3
Sweet	Achievement	C	Cooperative	8	13	7.4
		D	Competitive	5		6.3
	Ingenuity	C	Cooperative	8	13	7.3
		D	Competitive	5		6.5

6 Discussion

6.1 Types: Cooperative and competitive

Based on the results, the reason for the difference in achievement between the cooperative and competitive types is attributed to the attitude required of children to engage in programming. In the competitive type, participants must persistently strive to meet all provided conditions toward the specific goal of obtaining a reward. However, in competitive programming, children should perform persistent trial-and-error to satisfy all the provided conditions toward the specific goal of obtaining a reward. In the cooperative program, emphasis was placed on communication with the pair, and it is possible that the achievement scores were lower because more children completed the program only halfway, instead of completing it to fulfill all the conditions. We believe that this is because of the undefined roles of the driver and navigator in the general pair programming [20].

In the previous analysis, the competitive type outperforms the cooperative type. The difference in performance can be attributed to the different attitudes toward programming fostered by each type. Specifically, the competitive type required participants to fulfil all the stipulated requirements with the explicit goal of earning a reward. However, the cooperative type emphasizes collaboration through communication and requires participants to engage with their pairs. Although this approach facilitates dialogue, it may have strayed from the primary objective of fulfilling all predefined requirements. Consequently, many participants may have reached the midpoint of the required tasks. This phenomenon could be attributed to the fact that participants were not assigned roles, such as drivers or navigators, which is common in pair programming. Thus, participants prioritized discussion over implementation, resulting in less time being allocated to actual coding [21].

This illustrates that both educational methods are effective in programming education: the competitive method, in which specific objectives are set, and the cooperative method, in which students engage in collaborative work under well-defined roles.

6.2 Favorite Taste and Achievement

Children with a preference for spicy flavors tended to achieve higher levels of performance in competitive settings, whereas those favoring sweet flavors generally achieved better performance in cooperative settings. Charles synthesized research demonstrates a correlation between taste preferences and personality traits [22]. Traditionally, taste is classified using two primary methods: the Puise method, which includes four basic tastes (sweet, salty, bitter, and sour) and a broader classification that adds umami, resulting in five basic tastes [23]. In this study, spiciness was selected as the primary focus alongside sweet, salty, and sour flavors because of its frequent examination in the context of personality traits despite not being classified as taste. Umami was excluded from the analysis owing to substantial individual variability in its perception, while bitterness was omitted because it did not align with the study's objective of exploring personality traits through children's preferred tastes.

Students who prefer spicy tastes often exhibit higher risk-seeking behavior and greater sensitivity to rewards [24][25]. This tendency suggests that children with a preference for spicy tastes are particularly motivated by specific rewards within the competitive type, which in turn may lead to enhanced performance. Conversely, analysis utilizing the FFM indicates that a preference for sweet tastes correlates positively with traits, such as openness and diplomacy [26][27]. These personality traits are likely to enhance performance in the cooperative type by fostering increased sociability and a willingness to engage in new experiences.

7 Limitation and Future Work

There are three main limitations in the experimental design.

First, there was a notable disparity in the distribution of grade levels among different types of participants. Predominantly, fifth and sixth graders were involved in the competitive type, likely possessing a more advanced understanding of programming concepts, such as 'while' and 'if' statements compared to the fourth graders who were more represented in the cooperative type. This variation in age and associated cognitive abilities may have influenced the observed discrepancies in average academic achievement. In the competitive type, the average scores were as follows: 3.83 for 4^{th} graders (three students), 11.01 for 5^{th} graders (11 students), and 9.21 for 6^{th} graders (seven students). Conversely, in the cooperative type, no 6^{th} graders were present, and there were three 5^{th} graders and twenty 4^{th} graders. The uneven age distribution and associated developmental differences in understanding programming logic could potentially skew achievement outcomes in favor of older students.

Second, a sustained assessment over time is critical to evaluate whether the pedagogical approaches advocated in this study genuinely aid in the retention of programming knowledge. Considering that this study was conducted in a single session, it lacked the longitudinal component necessary to assess whether students could effectively internalize and reapply the taught programming concepts.

Third, the absence of a specialized personality ascertainment survey represents a significant shortcoming of this study. Such surveys are essential to accurately determine whether children's preferences are indicative of their underlying personality traits. However, because of time constraints, this critical component could not be implemented. Consequently, any conclusions derived from the analyses that link preferences with personality traits must be interpreted with caution.

Future studies should involve larger cohorts by including multiple classes within the same school and extending the duration of the study to the entire semester. Such an expansion and prolongation would enable a more thorough investigation of the long-term retention of knowledge facilitated by different instructional methods. Additionally, conducting a separate personality survey would allow researchers to assess rigorously whether the choice of instructional method, mediated by individual preferences, accurately reflects effective teaching strategies.

8 Conclusion

This study investigated the correlations between programming instruction methods and various preferences of elementary school students to distinguish between the distinct impacts of cooperative and competitive instructional techniques. Specifically, children with a preference for brilliant food demonstrated higher achievement scores under competitive instructional conditions. Because responses can be obtained more quickly than with traditional time-consuming methods, ascertaining personality traits through preferences is practical for one-off educational events, such as summer camps. In the future, we aim to verify the relationship between programming ability and preferences, and conduct more personalized programming education research.

Acknowledgements

This study was supported by Toyo University's Top-Priority Research Program.

References

[1] World Economic Forum , "These are the top 10 job skills of tomorrow – and how long it takes to learn them", Apr. , 24; https://www.weforum.org/agenda/2020/10/top-10-work-skills-of-tomorrow-how-long-it-takes-to-learn-them/

[2] International Labour Organization, "Skills shortages and labour migration in the field of information and communication technology in Canada, China, Germany, India, Indonesia, Singapore and Thailand", Sep. 2020: https://www.ilo.org/global/docs/WCMS_755663/lang--en/index.htm

[3] V. Garneli, M. N. Giannakos, K. Chorianopoulos, “Computing education in K-12 schools: A review of the literature”, IEEE IEEE Conf. Global Education Engineering (EDUCON 15), 2015, pp.543-551.

[4] Y. Yang, C. Augustine, C. Werts, Y. Yang, J. Sherwin, “Build Bridges between Conversation, Story, Visual Programming, and Text-based Programming in K-8 Education”, IEEE Conf. Information and Education Technology (ICIET 23), 2023, pp.226-231.

[5] H. Tsukamoto et al, “Programming education for primary school children using a textual programming language”, IEEE Conf. Frontiers in Education(FIE 15), 2015, pp.1-7.

[6] P. Korber, R. Motschnig, “The effects of pair-programming in introductory programming courses with visual and text-based languages”, IEEE Conf. Frontiers in Education(FIE 21), 2021, pp.1-9.

[7] M. Gómez, M. Moresi, L. Benotti, “Text-based Programming in Elementary School: A Comparative Study of Programming Abilities in Children with and without Block-based Experience”, ACM Conf. Innovation and Technology in Computer Science Education(ITiCSE 19), 2019, pp.402-408.

[8] C. Yongqiang, W. Xiaojun, Q. Chengbin, “Computer Programming Education for Primary School Students”, IEEE Conf. Computer Science & Education (ICCSE 18), 2018, pp.1-5.

[9] J. Moreno-León, G. Robles, “Code to learn with Scratch? A systematic literature review”, IEEE Conf. Global Education Engineering (EDUCON 16), 2016, pp.150-156.

[10] A. Issaee, R. Motschnig, O. Comber, “Pair- versus solo-programming of mini-games as a setting for learning to program: An Action Research approach”, IEEE Conf. Frontiers in Education(FIE 21), 2021, pp.1-9.

[11] I. R McChesney, “Three Years of Student Pair Programming: Action Research Insights and Outcomes”, ACM Technical Symposium on Computing Science Education(SIGSCE 16), 2016, pp.84-89.

[12] Y. Si, X. Chen, W. Guo, B. Wang, “The Effects of Cooperative and Competitive Situations on Statistical Learning”, Brain Science; doi:10.3390/brainsci12081059.

[13] E. Zakaria, L. C. Chin, M. Y. Daud, “The Effects of Cooperative Learning on Students’ Mathematics Achievement and Attitude towards Mathematics”, Social Sciences, doi: 10.3844/jssp.272.275

[14] C. Lin, S. Huang, J. Shih, A. Covaci, G. Chinea, “Game-Based Learning Effectiveness and Motivation Study between Competitive and Cooperative Modes”, IEEE Conf. Advanced Learning Technologies(ICALT 17), 2017, pp.123-127.

[15] S. Kulturel-Konak, M. L. D’Allegro, S. Dickinson, “Review of Gender Differences in Learning Styles: Suggestions for STEM Education”, Contemporary Issues in Education Research(CIER 11), vol.4, no.3, 2011, pp.9-18.

[16] C. Montuori, L. Ronconi, T. Vardanega, B. Arfé, "Exploring Gender Differences in Coding at the Beginning of Primary School", Frontiers in Psychology, vol. 13, 2022

[17] N. Salleh, E. Mendes, J. C. Grundy, G. St J Burch, "An Empirical Study of the Effects of Conscientiousness in Pair Programming using the Five-Factor Personality Model", ACM/IEEE Int'l Conf. Software Engineering, vol.1, 2010, pp.577-586.

[18] N. Salleh, E. Mendes, J. Grundy, "The effects of openness to experience on pair programming in a higher education context", IEEE-CS Conf. Software Engineering Education & Training(CSEE&T 11), 2011, pp.149-158.

[19] J. Jue, J. Hee Ha, "Exploring the relationships between personality and color preferences", Frontiers in Psychology, vol.13, 2022

[20] S. Mendolia, I. Walker, "The effect of personality traits on subject choice and performance in high school: Evidence from an English cohort", Economics of Education Review, vol.43, 2014, pp.47-65.

[21] L. Werner et al, "Pair programming for middle school students: does friendship influence academic outcomes?", ACM technical symposium on Computer science education, 2013, pp.421-426.

[22] C. Spence, "What is the link between personality and food behavior?", Current Research in Food Science, vol.5, 2022, pp.19–27.

[23] W. S. U. Roland, L. Pouvreau, J. Curran, F. van de Velde, P. M. T. de Kok, "Flavor Aspects of Pulse Ingredients", CEREAL CHEMISTRY, vol.94, 2016, pp.58-65.

[24] X. Wang, L. Geng, J. Qin, S. Yao, "The potential relationship between spicy taste and risk seeking", Judgment and Decision Making, vol. 11, no. 6, 2016, pp. 547–553.

[25] S. Spinelli et al, "Personality traits and gender influence liking and choice of food pungency", Food Quality and Preference. vol. 66, 2018, pp. 113–126.

[26] A. J. Saliba, K. Wragg, P. Richardson, "Sweet taste preference and personality traits using a white wine", Food Quality and Preference, vol. 20, 2009, pp. 572–575.

[27] N. W. Fitch, M. O. Gaylor, "DOES INDIVIDUAL PERSONALITY TYPE INFLUENCE DIETARY TASTE PREFERENCE AMONG SOUTH DAKOTA STATE UNIVERSITY (SDSU) STUDENTS ?", Proc. of the South Dakota Academy of Science, vol. 92, 2013.