Computational Thinking Level of Student in Statistics Using Computational Thinking Scale

Harsa Wara Prabawa[1]*, Rizky Rosjanuardi[2], Elah Nurlaelah[2]

[1] Computer Science Education, Universitas Pendidikan Indonesia, Bandung, Indonesia
[2] Mathematics Education, Universitas Pendidikan Indonesia, Bandung, Indonesia
harsawara@upi.edu

Abstract. A review of the literature on Computational Thinking has revealed the importance of computational thinking (CT) as an effective approach in problem-solving, especially when associated with the presence of technology as a means to formulate problem solutions. Using the Computational Thinking Scale (CTS), the purpose of this study is to map the development of students' CT as an impact of learning statistics. This research is descriptive. To determine the validity of the scale, a confirmatory factor analysis (CFA) calculation was conducted. As well as the power of difference, internal consistency, and the level of stability. The Computational Thinking Scale (CTS) was determined as a valid and reliable instrument to determine the CT ability of undergraduate students, based on the results of data processing. Students' ability in CT is in the medium range, with "Problem Solving" and "Creativity" as the most dominant factors. In contrast, "Algorithmic Thinking" is a factor that needs more attention. This indicates that students should be exposed to more non-routine and non-procedural problem-solving case studies, to be more familiar with algorithmic patterns in problem-solving.

Keywords: Computational Thinking, Computational Thinking Scale (CTS), Confirmatory Factor Analysis (CFA)

Introduction

In diverse learning research and practice, computational thinking (CT) has become a popular topic over the past decade. There are thousands of entries on search engines that map CT in various contexts: definition [1-7]; instructional implementation [1, 3, 8]. Some of them are even concerned with the process of knowledge transfer from a particular discipline to the computational process [9-11]; the field of mathematics [3, 12-13] dominates this issue. Several of these entries imply that CT is related to coding, while others contend that CT is more about programming than coding [9].

Although there is no standardized definition of CT, its existence and utility in the problem-solving process are not in question [2, 4-5]. CT has boundaries and a general framework that should be regarded for future growth [14]. Despite this, CT can still be regarded as a valid foundational skill that is not exclusive to the computer field but

F. Khoerunnisa et al. (eds.), *Proceedings of the 9th Mathematics, Science, and Computer Science Education International Seminar (MSCEIS 2023)*, Advances in Social Science, Education and Humanities Research 860,
https://doi.org/10.2991/978-2-38476-283-5_9

applies to other disciplines as well. This point with domain-specific and domain-general terminology [15]. Domain-specific refers to the utilization of domain-specific knowledge or skills that are required for systematically solving problems. Programming is, of course, a domain-specific topic in the context of scientific disciplines, as is the material examined specifically in the field of computer science. Domain-general refers to the skills required to solve problems systematically in daily life or across domains. In this context, CT is viewed more as a thinking process framework.

With the growing research focus on CT, including the possibility of using it as a representation of learning performance or learning outcomes [15], an intriguing research question emerges regarding how to assess the growth and development of CT. Some studies evaluate CT using programming-oriented tasks or tests of conceptual understanding [16]; others use robotics programming and reasoning for everyday events [17]. As a guide for problem formulation, previous research proposed a five-component framework (syntax, data, algorithm, representation, and efficient and effective) [17]. Some of them utilized conceptual case study problems [18-19] and even Bebras Challenge data [20].

How the attainment of CT is evaluated is, without a doubt, another significant area of research in CT development [14]. From the numerous definitions of CT, this article positions CT as a process of problem-solving thought. Situational analysis is frequently the impetus and driving force behind problem-solving efforts. Text discourse analysis, subject assessments, job analysis, graph analysis, and behavior analysis are therefore the most likely evaluation methods to be used [14], which necessitate a validated objective framework to evaluate CT. One of the obstacles to consider in evaluating CT is the lack of consensus on assessment criteria and appropriate instructional feedback [21]. the Computational Thinking Scale (CTS) as a reference framework for gauging CT development by basing it on the problem at hand [14].

CTS is a measurement tool devised and utilized to assess CT. Researchers have developed at least a few CT measurement instruments, including:

- Computational Thinking Self-Efficacy Scale [22] consists of 18 items that measure four factors: reasoning, abstraction, decomposition, and generalization. This scale was created to predict CT activities.
- The Computational Thinking Scale [23] is comprised of 42 items that measure four factors: problem-solving, cooperative learning and critical thinking, creative thinking, and algorithmic thinking. The CT of secondary school students is measured by this scale.
- Computational Thinking Scale [24] consists of 30 items that measure three factors: robotic and software programming, computational thinking, professional development, and career planning. This instrument was designed to measure CT in primary school pupils.
- Computational Thinking Scale [25]: This scale is comprised of 22 items that measure five factors: abstraction, decomposition, pattern, algorithm, and generalization. This instrument was devised to assess CT in secondary school students.

- Computational Thinking Scale [15]: This scale is comprised of 19 items that measure five factors: abstraction, decomposition, algorithmic thinking, evaluation, and generalization. This scale was devised to measure CT among secondary school students.

The purpose of this study is to describe the computational thinking ability of students by adapting CTS, especially at the first level of students in college. Of course, this research aims to prove or test what has been claimed by previous research, which is more centered on learning at the primary and secondary school levels. It is hoped that this scale can provide significance for measuring CT, especially at the higher education level. To achieve this goal, the data will be analyzed using the confirmatory factor analysis (CFA) method, adopting the opinion of for the determination of model acceptance criteria and analysis results [26].

2 Methodology

In addition to being a descriptive study, this research examines the adaptation of the CTS. In this context, the CT of students has been attempted to be determined based on studies conducted by previous researchers. The CTS utilized in this study can be downloaded at https://csedresearch.org/wp-content/uploads/Instruments/Computing/PDF/ComputationalThinkingScales.pdf.

This investigation included 148 first-year Computer Science Education students who enrolled in a statistics course. The CTS, comprised of 29 items and five factors, was adapted to capture data for this study [25]. As explained in the introduction, to measure the CT skills of first-year Computer Science Education students taking the Statistics course this scale underwent adjustments, because of the confirmatory factor analysis (CFA) conducted, resulting in only 16 items and four factors. The CFA performed on the data and the parameters obtained showed an acceptable fit for this four-factor structure.

The researcher utilized the JASP 0.17.3 program to process CFA data for this study. CFA is used to determine whether indicator variables can be used to corroborate a factor. The CFA model can be evaluated using the following four criteria: (1) model convergence and acceptable range of parameter estimates; (2) fit indices; (3) significance of parameter estimates and related diagnostics; and (4) measurement invariance across multiple samples [26].

On a Likert scale, each item was rated as follows: never (1), rarely (2), sometimes (3), generally (4), and always (5). The scores obtained from the answers provided by the students to the five Likert-type scales do not present a standardized picture due to the different item numbers within the factors. The derived raw scores on the Likert scales were then transformed into standardized scores, with the lowest measure being 20 and the highest measure being 100, for further data processing. To convert raw scores to standard scores, it is necessary to multiply the sum of the item scores for each factor by 20, then divide by the number of items for that factor.

The leveling refers to the equivalence of the scores obtained from the subscales by using the following calculation: if the acquisition of the conversion value is < ($\bar{x}$ -

standard deviation), then it belongs to the low level; if it is > ($\bar{x}$+ standard deviation), then it belongs to the high level. Data that did not fall into the low or high categories were classified as medium.

3 Result and Discussion

3.1 CTS Validity and Reliability Measurement

The initial CFA results for the CTS variable are shown in Table 1. In general, Table 1 demonstrates that the measurement model is not appropriate. The parametric fit p-value, GFI, RMSEA, NFI, IFI, CFI, and TLI do not conform to the established criteria. Since the model is not yet optimal, the researcher modifies it to acquire a superior model. The model is modified by omitting items with factor loadings of less than 0.30, including CR five items for the factor creativity, two items for the factor algorithmic thinking, one item for the factor cooperative, and all items for the factor critical thinking. The removal of items with factor loading below 0.3 was based on the opinion of which states that the weakest acceptable factor loading is 0.30 [26]. Therefore, these elements will be absent from the final model. No items were taken away for the problem-solving factor. Table 2 displays the final model's accuracy parameters after the elimination of three items.

Table 1. CTS Accuracy Before Modification

Category	Parameter Fit	Output	Criteria	Description
Absolute Fit	Chi-square P-Value	< 0.01	≥ 0.05	no fit
	Goodness of fit index (GFI)	0.979	≥ 0.90	fit
	Root mean square error of approximation (RMSEA)	0.110	≤ 0.08	no fit
Incremental fit	Normed fit index (NFI)	0.525	≥ 0.90	no fit
	Incremental fit index (IFI)	0.633	≥ 0.90	no fit
	Comparative fit index (CFI)	0.625	≥ 0.90	no fit
	Tucker-Lewis Index (TLI)	0.585	≥ 0.90	no fit
Parsimonious fit	Parsimonious Normal Fit Index (PNFI)	0.475	0.60 – 0.90	no fit

Table 2. CTS Accuracy after Modification

Category	Parameter Fit	Output	Criteria	Description
Absolute Fit	Chi-square P-Value	0.355	≥ 0.05	fit
	Goodness of fit index (GFI)	0.988	≥ 0.90	fit
	Root mean square error of approximation (RMSEA)	0.054	≤ 0.08	fit
Incremental fit	Normed fit index (NFI)	0.960	≥ 0.90	fit
	Incremental fit index (IFI)	0.953	≥ 0.90	fit
	Comparative fit index (CFI)	0.952	≥ 0.90	fit
	Tucker-Lewis Index (TLI)	0.942	≥ 0.90	fit

Category	Parameter Fit	Output	Criteria	Description
Parsimonious fit	Parsimonious Normal Fit Index (PNFI)	0.702	0.60 – 0.90	fit

Table 2 displays the modified model's ultimate form. In the final model, the p-value for chi-squared is greater than 0.05. There is no significant difference between the ideal model and the proposed model based on observational data, indicating that the model is appropriate. The GFI, NFI, CFI, TLI, IFI, and PNFI values also meet the criteria set to obtain model fit. Consequently, this final model is fit, which indicates that the proposed model corresponds to the empirical data. Figure 1 depicts the complete model and loading factor for each variable in the final model.

Following model testing and factor load analysis, the reliability test is conducted. The internal consistency of an instrument, which can be measured based on the degree of item homogeneity, can be interpreted as reliability. Construct reliability (CR) and average variance extracted (AVE) comprise the reliability test in CFA analysis [27]. If the CR value is less than or equal to 0.70, the item is categorized as reliable. In the meantime, the CR value ranges between 0.6 and 0.7, indicating acceptable reliability if the indicator's factor load meets the criteria. Using the AVE estimate, internal consistency can also be evaluated. The suggested AVE value is greater than 0.5 [27]. Based on the data in Table 3, it is known that all CTS instrument factors and dimensions have high reliability.

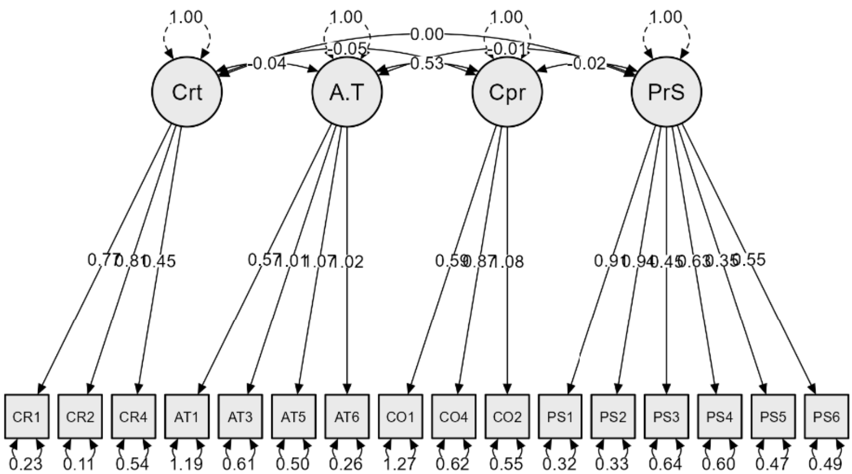

Fig. 1. CTS Measurement Model

Based on the calculation of CR and AVE on the CTS factor loading, it is evident that the CT factors and dimensions in the form of creativity, algorithmic thinking, cooperation, and problem-solving have convergent validity. Convergent validity quantifies the extent to which a measurement is positively correlated with other measurements that measure the same construct. Consequently, indicators of a construct should converge or share a significant proportion of variance. If an instrument satisfies the convergent validity requirements of factor loading ≥ 0.5, composite reliability value ≥ 0.7, and average variance extracted (AVE) ≥ 0.5 [27], it is considered to have convergent validity.

The results of CFA calculations should also reveal the discriminant validity of the model. Validity is a measure of the extent to which a construct differs from other constructs, which means that a latent variable must explain the variance of its indicators better than other latent variables. Consequently, the factor loading of the indicator on the latent variable in question must be greater than the factor loadings on all other latent variables. One method for evaluating discriminant validity is to compare the AVE to the square of the correlation between the two constructs. Discriminant validity is obtained when the square root of the AVE is greater than the correlation between constructs [26]. Based on the data presented in Table 4, it is known that the four dimensions of CTS possess discriminant validity. The number hacked below the main diagonal represents the correlation coefficient (r), the number hacked on the main diagonal represents the AVE value, and the number hacked above the main diagonal represents the correlation square (r^2).

Based on the results of the CFA analysis, it can be concluded that CTS is primarily influenced by four dimensions: creativity, algorithmic thinking, cooperative, and problem-solving. The existence of critical thinking does not contribute to the model. This is believed to be the consequence of the cooperative dimension's predominance, which allows it to compensate for individual limitations in problem-solving. This condition can be interpreted as proof that the CTS model has evidence of internal structural validity. In addition, convergent and discriminant validity evidence supports the CTS model. The CTS model has a relatively high degree of dependability, allowing it to produce repeatable results that are relatively consistent.

Table 3. CR and AVE values in CTS

Indicator	λ	Error	λ^2	CR	AVE
Creativity [Crt]					
CR1	0.851	0.275	0.724	0.821	0.616
CR2	0.923	0.149	0.852		
CR4	0.523	0.727	0.274		
Total	2,297	1,151	1,850		
Alg. Thinking [A.T]					
AT1	0.564	0.784	0.318	0.851	0.594
AT3	0.791	0.375	0.626		
AT5	0.833	0.307	0.694		
AT6	0.894	0.201	0.799		
Total	3,082	1,667	2,437		
Cooperative [Cpr]					
CO1	0.562	0.627	0.316	0.764	0.525
CO2	0.823	0.322	0.677		
CO4	0.742	0.449	0.551		

Indicator	λ	Error	λ^2	CR	AVE
Total	2,127	1,398	1,544		
Problem-solving [PrS]					
PS1	0.847	0.283	0.717	0.856	0.505
PS2	0.852	0.275	0.726		
PS3	0.591	0.629	0.349		
PS4	0.630	0.503	0.397		
PS5	0.557	0.611	0.310		
PS6	0.615	0.522	0.378		
Total	4.092	2.823	2.878		

Table 4. AVE Values and Shared Variance Estimates

Variable	item	1	2	3	4
Creativity [Crt]	3	0.616	0.001	0.002	0.000
Alg. Thinking [A.T]	4	0.036	0.594	0.280	0.000
Cooperative [Cpr]	3	0.049	0.529	0.525	0.000
Problem-solving [PrS]	6	0.004	0.011	0.019	0.505

3.2 Level of Students' Computational Thinking

As shown in Table 5, students' scores in CT ranged from 55.79 to 74.68, with a mean of 65.24. The scores were obtained with 16.89% of students possessing a high skill level, 68.92% possessing a medium skill level, and 13.51 % possessing a low skill level. The level of computational thinking is moderate. The factor with the greatest average score is "creativity" (=81.32), whereas the factor with the lowest average score is "algorithmic thinking" (=47.11). In contrast, in the high group, the skill levels of "algorithmic thinking" and "cooperative" received equal proportions (16.89%), and "problem-solving" received the lowest proportion (6.08%). Based on this, it can be concluded that "algorithmic thinking" and "cooperative thinking" are the most prevalent student skill levels, while "problem-solving" is the least prevalent.

Table 5. Students' Computational Thinking Skill Level

Factor	n	$\bar{x}$	St.Dev.	Min.	Max.	Level (f/%)		
						Low	Md.	High
Creativity	148	81.32	14.96	66.35	96.28	11.49	72.30	15.54
Alg. Thinking		47.11	20.05	27.06	67.16	15.54	66.89	16.89
Cooperative		56.33	19.97	36.35	76.30	16.22	66.22	16.89
Problem-solving		73.74	14.26	59.48	88.01	12.84	80.41	6.08
Computational Thinking		65.24	9.44	55.79	74.68	13.51	68,92	16.89

In general, students' CT is moderate, with "creativity" and "problem-solving" being the most influential factors. This confirms the findings of some previous researchers that there is a reciprocal relationship between creativity and computer science, particularly CT. On the one hand, it has been demonstrated that computerized platforms and programming activities inspire creativity in the production of artifacts in fields such as art, graphic design, and music [28-30]. On the other hand, creativity is a catalyst for solving algorithmic problems, developing computational artifacts, and acquiring new knowledge [31, 32]. Digital learning platforms that promote programming, or CT, frequently provide opportunities to broaden creative expression and foster the growth of creative thought. Previous study revealed that "problem-solving" can be defined as a series of goal-directed cognitive operations [33]. This definition does not differentiate between sequences of actions that are known to achieve a goal and sequences of actions that must be performed when the means to achieve the goal are unknown [33]. The former is the consequence of experience, while the latter is the situation encountered by a novice. Then, Anderson referred to the experience as an automated activity required for problem-solving, whereas the other activity represented an initial attempt to solve the problem [34]. The presence of "creativity" and "problem-solving" in students can be interpreted as a worthwhile investment in the development of CT.

What merits attention is the low grade for the ability of "algorithmic thinking" when compared to other skills. The low proficiency in "algorithmic thinking" is likely due to students' limited comprehension of determining the optimal path from the problem state to the target state. Alan Perlis uses the term algorithmic to describe the quantitative analysis of how a person performs a task and classifies it as a fundamental thought process that everyone must master [9]. It can be concluded that if students possess this skill, they are more likely to possess other skills [14]. Consequently, it is not surprising that this ability received the lowest average score and one of the highest scores in the high group. This is due to the disparity in students' knowledge and comprehension of the problem, its formulation into solution stages, and the execution of the plan through computational activities.

4 Conclusion

CT is an indispensable component of the problem-solving procedure. Therefore, individuals must acquire and develop these skills throughout their educational experience. Specifically, it is stated that if we examine the above-mentioned factors (creative thinking, algorithmic thinking, critical thinking, problem-solving, and cooperation skills), then individuals will have these skills at school age, develop themselves, create opportunities to be able to adapt to technological developments more quickly and produce something more productive based on technology. This is why CT should be taught in elementary education so that students can transition this skill to other problem situations. Due to the prevalence of complex algorithms and problems in everyday life, developing these skills in school will be a significant advantage. Based on the findings of this study and the relevant literature, it is suggested that students frequently engage in activities designed to enhance their problem-solving and algorithmic thinking skills, particularly in the context of diverse materials and subjects.

References

1. Papert, S. Mindstorms: Children, Computers, and Powerful Ideas. Basic Books (2020).
2. Hoppe, H. U., Werneburg, S. Computational Thinking—More Than a Variant of Scientific Inquiry! In S.-C. Kong, & H. Abelson (Eds.), Computational Thinking Education, pp. 13-30. Springer Open, Singapore (2019).
3. Yadav, A., Mayfield, C., Ninger Zhou, S. H., Korb, J. T. Computational Thinking in Elementary and Secondary Teacher Education. ACM Transactions on Computing Education, **14**(1), (2014)
4. Aho, A. V. Computation and Computational Thinking. The Computer Journal **55**(7), 832-835 (2012).
5. Wing, J. M. Computational Thinking. Communications of the ACM, **49**(3), 33-35 (2006).
6. Denning, P. J. the Profession of IT Beyond Computational Thinking. Communications of the ACM, **52**(6), 28-30 (2009).
7. Zakwandi, R., Istiyono, E., Dwandaru, W. S. B. A two-tier computerized adaptive test to measure student computational thinking skills. Education and Information Technologies **29**(7), 8579 – 8608 (2024).
8. Kafai, Y., Proctor, C., Lui, D. From Theory Bias to Theory Dialogue: Embracing Cognitive, Situated, and Critical Framings of Computational Thinking in K-12 CS Education. In: The 2019 ACM Conference on International Computing Education Research, pp. 101-109. ICER-ACM, (2019).
9. Lodi, M., Martini, S. Computational thinking, between Papert and Wing. Science & Education, **30**, 883-908 (2021).
10. Robins, A. V., Margulieux, L. E., Morrison, B. B. Cognitive Sciences for Computing Education. In S. A. Fincher, & A. V. Robins (Eds.), The Cambridge Handbook of Computing Education Research (pp. 231-275). Cambridge University Press (2019).
11. Ioannidou, A., Bennett, V., Repenning, A., Koh, K. H., Basawapatna, A. Computational Thinking Patterns. Annual Meeting of the American Educational Research Association. New Orleans, (2011).
12. Kallia, M., Borkulo, S. P., Drijvers, P., Barendsen, E., Tolboom, J. Characterizing computational thinking in mathematics education: a literature-informed Delphi study. Research in Mathematics Education. (2021)
13. Mol, L. D. Some reflections on mathematics and its relation to computer science. In A. Adamatzky (Ed.), Emergence, Complexity, and Computation: Tribute to Maurice Morgenstern (pp. 75-102). Switzerland: Springer (2015).
14. Korkmaz, Ö., Bai, X. Adapting Computational Thinking Scale (CTS) for Chinese High School Students and Their Thinking Scale Skills Level. Participatory Educational Research (PER), **6**(1), 10-26 (2019).
15. Tsai, M.-J., Liang, J.-C., Hsu, C.-Y. The Computational Thinking Scale for Computer Literacy Education. Journal of Educational Computing Research, **59**(4), 579-602 (2021).
16. Román-González, M., Moreno-León, J., Robles, G. Combining Assessment Tools for a Comprehensive Evaluation of Computational Thinking Interventions. In S.-C. Kong, & H. Abelson (Eds.), Computational Thinking Education (pp. 79-98). Springer, Singapore (2019).

17. Chen, G., Shen, J., Barth-Cohen, L., Jiang, S., Huang, X., Eltoukhy, M. Assessing elementary students' computational thinking in everyday reasoning and robotics programming. Computers & Education, 162-175 (2017).
18. Bavera, F., Quintero, T., Daniele, M., Buffarini, F. Computational Thinking Skills in Primary Teachers: Evaluation Using Bebras. In: 25th Argentine Congress of Computer Science, pp. 405-415, Río Cuarto, (2019).
19. Matsuzawa, Y., Murata, K., Tani, S. Multivocal Challenge Toward Measuring Computational Thinking. In: Open Conference on Computers in Education (OCCE), pp. 56-66. Linz, Austria (2018).
20. Araujo, A. L., Andrade, W. L., Guerrero, D. D., Melo, M. R. How Many Abilities Can We Measure in Computational Thinking? A Study on Bebras Challenge. In: The 50th ACM Technical Symposium on Computer Science Education, pp. 545-551. Association for Computing Machinery (2019)
21. Alves, N. D., Wangenheim, C. G., Hauck, J. C. Approaches to Assess Computational Thinking Competences Based on Code Analysis in K-12 Education: A Systematic Mapping Study. Informatics in Education, **18**(1), 17-39 (2019).
22. Kukul, V., Karataş, S. Computational Thinking Self-Efficacy Scale: Development, Validity and Reliability. Informatics in Education, **18**(1), 151-164 (2019).
23. Yağcı, M. A valid and reliable tool for examining computational thinking skills. Education and Information Technologies, **24**(2), 1-23 (2019).
24. Ertugrul-Akyol, B. Development of Computational Thinking Scale: Validity and Reliability Study. International Journal of Educational Methodology, **5**(3), 421-432 (2019).
25. Korkmaz, Ö., Çakir, R., Özden, M. Y. A validity and reliability study of the computational thinking scales (CTS). Computers in Human Behavior, **72**, 558-569 (2017).
26. Netemeyer, R. G., Boles, J., Mcmurrian, R. C. Development and Validation of Work-Family and Family-Work Conflict Scales. Journal of Applied Psychology, **81**(4), 400-410 (1996).
27. Hair, J. F., Black, W. C., Babin, B. J., Anderson, R. E. Multivariate Data Analysis, 7th ed. Pearson Prentice Hall (2010).
28. Israel-Fishelson, R., Hershkovitz, A., Eguíluz, A., Garaizar, P., Guenaga, M. The Associations Between Computational Thinking and Creativity: The Role of Personal Characteristics. Journal of Educational Computing Research, **58**(8), (2020).
29. Lau, K. W., Lee, P. Y. The use of virtual reality for creating unusual environmental stimulation to motivate students to explore creative ideas. Interactive Learning Environments, **23**(1), 3-18 (2015).
30. Mishra, P., Yadav, A. Rethinking Technology & Creativity in the 21st Century: Of Art & Algorithms. TechTrends, **57**(3), 10-14 (2013).
31. Kong, S.-C. Components and methods of evaluating computational thinking for fostering creative problem-solvers in senior primary school education. In S.-C. Kong (Ed.), Computational Thinking Education, pp. 119-141. SpringerLink (2019).
32. Pérez-Poch, A., Olmedo-Torre, N., Sánchez-Carracedo, F., Salán-Ballesteros, M. N., López-Álvarez, D. On the influence of creativity in basic programming learning at a first-year Engineering course. International Journal of Engineering Education, **32**, 2302-2309 (2016).
33. Robertson, S. I. Problem Solving. Taylor & Francis, Philadelphia (2001).
34. Anderson, J. R. Cognitive Psychology and its Implications. 9th ed. Worth Publishers - Macmillan Learning, New York (2020).