Computational Thinking Test (CTT) for Middle School Students*

Kaan Batı[1]

Abstract: The aim of this research is to develop a valid and reliable Computational Thinking Test (CTT) that can be used at the eighth grade level. The computational thinking skills defined in the literature were identified and then the problematic situations which could measure these skills were established. Our CTT included 10 problem cases which were constructed in connection with the subject areas of science and mathematics education programs. The test was sent to five field experts and they were asked to evaluate the construct validity and content validity of the test items. The test, revised according to the expert feedback was then administered to 110 eighth-grade students from a state secondary school. Analysis results indicate that CTT is a valid and reliable (McDonald's ω = 0,79) test that can successfully measure the computational thinking skills of 8th grade students.

Keywords: computational thinking, computational thinking test

DOI: 10.29329/mjer.2018.138.6

* Bu makalenin bulguları "ECER 2017 Copenhagen (European Conferences of Educational Research - 2017)" kongresinde aynı isimle sunulmuştur.

[1] Arş. Gör. Dr., Hacettepe Üniversitesi, Egitim Fakültesi, Matematik ve Fen Bilimleri Egitimi Bölümü, Fen Bilgisi Ögretmenligi Ana Bilim Dali, kaanbati@hacettepe.edu.tr

INTRODUCTION

Education programs are being revised in order to be able to raise individuals who can effectively use and develop information communication technologies (ICT), with the increasing influence of information communication technologies on human life and country economies. At the forefront of the planned skills to bring individuals into this field is the ability to perform computational thinking. The essence of computational thinking is being like a computer scientist when faced with a problem. In a wider sense, computational thinking is formulating problems in a way that is appropriate for solving them with computing tools such as computers, logical arrangement and analysis of data, abstracting them as data models and simulations, and generating results with algorithmic thinking, in which finding possible solutions involves generalizing and transferring problem-solving processes to a wide range of problem types (Barr, Harrison, Conery, 2011; Wing, 2008). Individuals who have computational thinking skills use the general methods of mathematical thinking to solve a problem; and they think like an engineer by associating the problem with patterning and real life situations of a large, complex system of intelligence, mind, human behaviors, and scientific thinking (Wing, 2008). Computational thinking is not limited to computer software writing or coding knowledge and skills. It does not try to make people think like computers, but consists of scientific thinking, problem solving and communication as well.

Israel, Pearson, Tapia, Wherfel and Reese (2015) describe the notion of computational thinking as: formulating problems in a way that enables us to use a computer and other tools to help solve them; organizing and analyzing data logically; representing data through models and simulations; automating solutions through algorithmic thinking; identifying, diagnosing, analyzing and implementing possible solutions to achieve the most effective combinations of steps and resources, and generalizing this problem-solving process and passing it on to a wide range of problems. Weintrop, Beheshti, Horn, Orton, Jona, Trouille and Wilensky (2014) group computational thinking skills under four main categories as data and information skills, modeling and simulation skills, informational problem solving skills, and systematic thinking skills. The sub-dimensions of these categories are:

• Data and Information Skills: data collection, data generation, manipulation of data, analysis of data and visualization of data.

• Modeling and Simulation Skills: using information processing models to understand a concept, understanding how and why information processing models work, using information processing models to find and testing solutions, creating new models and expanding existing models.

• Computational Problem Solving Skills: catching and debugging errors, programming, choosing effective computational tools, measuring different approaches / solutions for a problem, developing modular information operational solutions, using problem solving strategies and creating abstracts.

• Systems Management Skills: Examining a system as a whole, understanding relationships within a system, thinking in levels and visualizing systems; defining, understanding and managing complexity.

Although computational thinking is defined theoretically, how these skills should be taught to children, how they can be measured, and which teaching approaches are the most important are among the questions to be answered (Grover & Pea 2013). The relevant literature reveals different approaches to measuring the development of the computational thinking skills. Werner, Denner Campe and Kawamoto (2012) developed a computer program (Fairy Assessment) for measuring the extent of the use of algorithmic thinking and modeling and concretization of computational thinking using the Alice program, and developed a computer game programming they practiced in class. Although the developed program may seem to measure the students' ability to perform computational thinking, they expressed the limitations of being independent of the scores obtained from different parts of the program. A similar research study was carried out by Webb (2011) and the training conducted with computer programs was evaluated. In this study, the researcher stated that the evaluation of the skills of the students with no prior knowledge and experience may cause some problems. Marshall (2011) used a video-based computer program to measure the computational thinking skills of middle school students, and reported that many learners could not complete the measurement process due to time and technology problems. Korkmaz, Çakır and Özden (2015) adapted the Computational Thinking Skills Level Scale (BDBD) to the middle school level. BDBD is a five-point Likert-type scale, and consists of 22 items that can be grouped under five factors. The scale measures students' emotional characteristics including expressions such as "I can do" and "I believe." However, this scale does not provide concrete data on how much students have actually gained these skills.

Assessing Computational Thinking

Computational thinking skills are generally considered in terms of computer use or coding process, and therefore, the most commonly preferred method of measuring these skills is the evaluation of students' competency in using robotics or software products, and their opinions during this process. However, valid and reliable tests that can measure students' computational thinking skills are not yet in sufficient quality and quantity. This might be because such skills are assessed by focusing mostly on student products, and that valid and reliable means of measurement are not yet established, as the ability of computational thinking is a relatively new field.

Some researchers have developed different kinds of tests considering the problems which are mentioned above. Csernoch, Biró, Máth, and Abari (2015) developed a two-stage test to measure the computational thinking skills of students in higher education. In the first stage, a self-assessment questionnaire and informatics knowledge questionnaire were applied for informational thinking and algorithmic thinking skills. In the second stage, a problem solving skills test was applied in the non-programming environment, testing the use of programming skills to solve high-level problems. Kim, Kim, and Kim (2013) proposed a method named *paper pen programming strategy* (PPS). This method allows students to create diagrams, cues, codes, symbols, tables, etc. for a logical idea or solution. Plotted, or printed by a pencil test, and the problem is analyzed five times, as a result of designing, constructing, implementing and correcting the algorithm. The properties distinguishing PPS from other methods are: (1) PPS not only focuses on the solution, but also supports communication skills related to presentation of solutions, (2) focuses on the programming language itself, as well as the designing process of the programming language, (3) when analyzing problems and using the idea of divergence such as brainstorming in the design of solutions, uses algorithmic thinking and convergent thinking in the process of debugging mistakes, (4) discovers and presents algorithmic solutions for everyday life problems, (5) allows students to use their own mental tools enabling them to think creatively, just like a computer scientist. Below are sample student solutions from PPS.

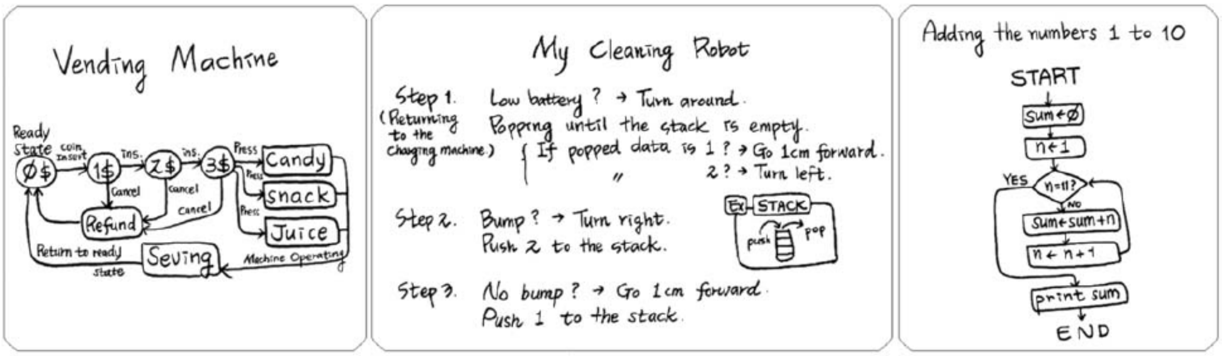


Figure 1: PPS examples (Kim, Kim & Kim, 2013)

The measurement tool developed by Jun, Han, Kim and Lee (2014) aims mainly to determine the literacy of information and communication technologies of primary school students. This test consists of three levels; the first level includes first and second classes, the second level third and fourth classes, and the third level includes questions at the fifth and sixth grade levels. Jun, Han, Kim and Lee (2014) stated that they have three legs of the test they have developed, these are basic concepts related to information communication technologies, contemporary skills expressing practical use of these concepts and cognitive (intellectual) skills such as problem solving skills. The examples for these questions are given in Figure 2.

The first applications of computational thinking and informatics activities in Turkey were held under the “Bilge Kunduz” forum in 2014. This activity was described in the 2015 Turkey report as follows:

Bilge Kunduz International Forum on Informatics and Computational Thinking (Bebras International Challenge on Informatics and Computational Thinking) was organized in Turkey for the first time in 2014. With the support of the volunteer coordinators who participated in the pilot application, 57 schools participated. 1,788 people successfully completed the event. In 2015, 257 schools participated in the Bilge Kunduz Event. 13,784 5th and 6th grade students completed the event successfully. This report was prepared in order to share the results of the event in 2015. (*http://www.bilgekunduz.org/wp-content/uploads/2016/01/bilgekunduz-rapor-2015.pdf*)

The activity explained above is called “The International Field Bebras Workshop (http://www.bebras.org). The questions in this activity were prepared at a workshop with the participation of the representatives of the member countries. Example for the questions is given in Figure 3.

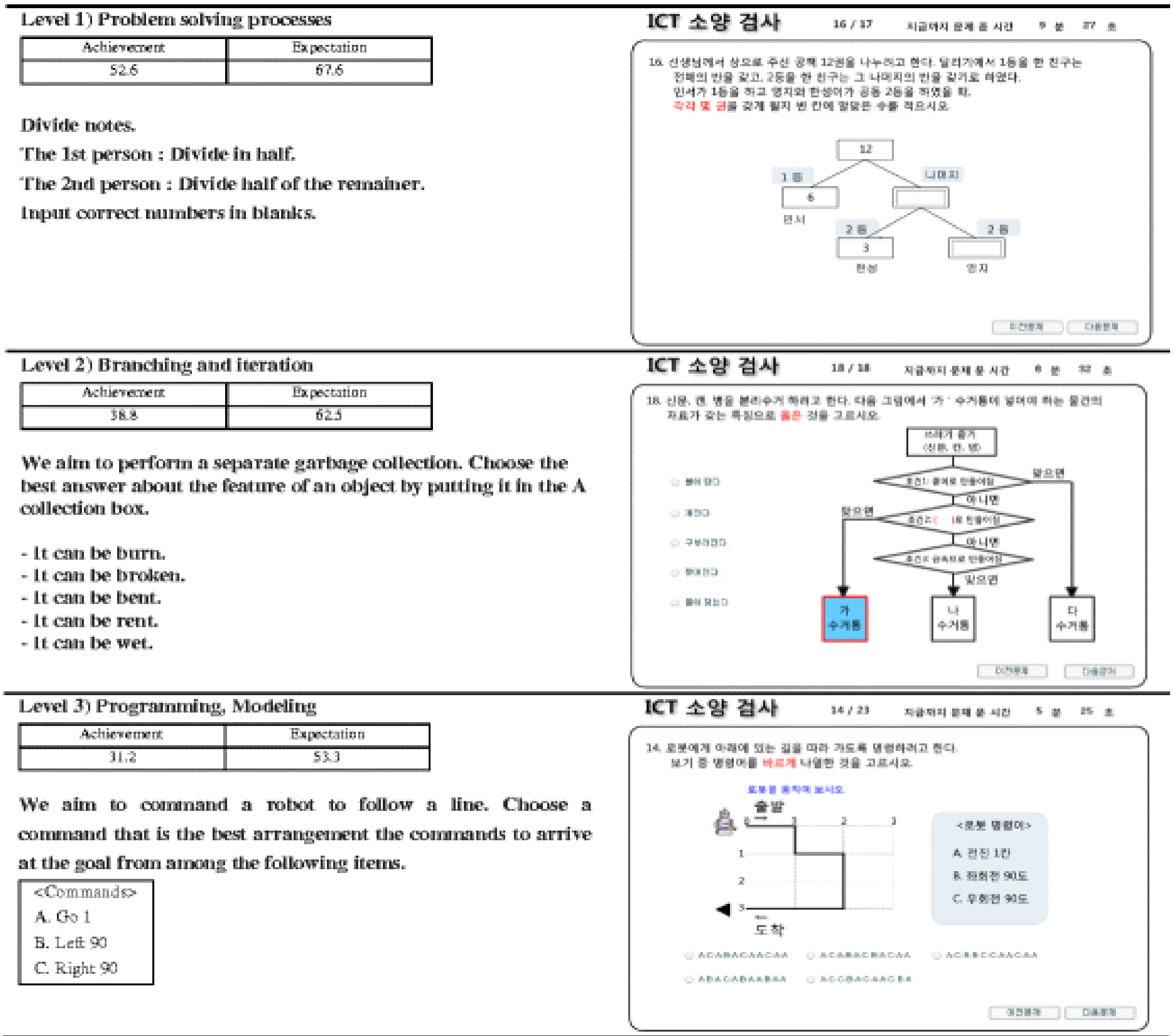
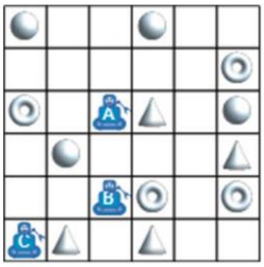
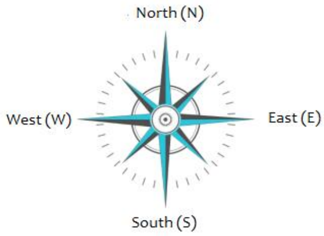


Figure 2: The examples of the questions from Jun, Han, Kim and Lee (2014)

In a store, A, B and C robots work together. These robots move in the same direction concurrently. Following the given directions, the robots are collecting the objects on the surface. For example, when the directions are given as "N, N, S, S, E", then robot A takes a cone, robot B takes a ring, and robot C takes a cone.





Question:

Which directions need to be followed in order for robots A, B and C to take sphere, ring and cone respectively?

A) *N, E, E, E,*
B) *N, E, E, S, E*
C) *N, N, S, E, N*
D) *N, E, E, S, W*

Figure 3: Bilge Kunduz Project example

Purpose of the Study

The mathematics and science education contribute at a significantly higher level to enhancing students' level of computational thinking skills than the other areas (Korkmaz et al 2015). Integrating computational thinking skills with the mathematics and science education is so important. For this reason, the research literature summarized above reveals that the tests for measuring the performance of the computational thinking need further improvements. Additionally, there is a clear need to integrate computational thinking skills into teaching programs by removing them from the context of computer programming and integrating them primarily with science and mathematics courses. The present study attempts to solve these deficiencies by developing a valid and reliable test to measure the computational thinking skills of secondary school students.

METHOD

This research is a scale development study. Within the scope of the study, primarily the ability of computational thinking as defined in the literature was determined. Following the literature review, the sub-dimensions of the informal thinking skills and the sub-skills to be measured were determined based on the categorizations made by Weintrop et al (2014). Within this scale development study, it was aimed to develop a measurement tool that could only measure computational problem solving skills and Systems Management Skills. Because data and knowledge management skills and modeling and simulation skills have been studied for a long time in science and mathematics education, many research studies have been done on how students can gain these skills and how these skills can be measured. However, the ability to perform computational problem solving and systems management is quite new at the secondary and high school levels, and in science and mathematics teaching.

Then the problem cases related to the subject areas of science and mathematics were created to measure these skills. Following the creation of the questions, they were sent to the measurement, science, and mathematics experts for their structure, scope and validity to be evaluated. Piloting was conducted with 28 eighth grade students who read the questionnaires again. The piloting was conducted in a state secondary school in the city of Ankara during the fall semester of 2016 - 2017. Students were asked to fill in voluntary participation forms. A total of 28 students were reached in the pilot study and students were asked to answer the questions in the test as well as to evaluate their readability and clarity. Even though the practice was expected to take one lesson, the practice lasted for two hours due to the fact that the students were not familiar with the questions and the difficulty of the problem situations were above the students' level. The test items were revised according to the data obtained from the pilot application. The following table shows the relationship of the ten questions to the determined skills obtained after ensuring the structure, scope, and appearance of the scale.

Sampling

The sample of this research was formed by 110 8th grade students studying in a state secondary school in Ankara, Turkey. Convenient sampling method was used in the sampling selection. They had not taken any coding or ICT technology courses before the application of the CTT. The descriptive statistics of the study group are presented in Table 1.

Table 1. Descriptive statistics

	N	Mean	Std. Dev.	Variance	Skewness	Kurtosis
Group	110	33.4091	16.63659	276.776	.234	-.330

In CTT there are ten questions and each question is ten points, accordingly, students can take a maximum of 100 points in the test. As seen in the descriptive analyses, the group mean score is 33.4, and standard deviation is 16.6, so students had difficulty in answering the questions. However, when we examine Table 1, it is clear that the skewness values of the study group are within normal limits. Yet, normality test results provide further evidence that the group is suitable for normal distribution. Statistical values of normality tests are given in Table 2.

Table 2. Normality test statistics

	Kolmogorov-Smirnov		
	Statistic	df	Sig.
Group	.055	110	.200

Table 2 shows that the normal distribution of the data obtained from the study group is appropriate and parametric statistical methods can be applied.

Validity and Reliability

CTT consists of 10 problem cases in total. While one question was asked about some problem cases, two or three questions were asked about some problem cases. For this reason, partial scoring method was used for scoring CTT. Fleiss kappa statistics were used for scoring reliability in terms of this reason. Fleiss Kappa is a statistic used to determine scorer safety if the statistic involves more than two scorers (Atılgan, Kan, & Dogan, 2007). The statistical values for the kappa scores obtained are given in Table 3.

Table 3. KAPPA statistics

		Value	Asymp. Std. Error[a]	Approx. T[b]	Approx. Sig.
Measure of Agreement (scorer 1 and 2)	Kappa	.186	.095	5.628	.000
Measure of Agreement (scorer 1 and 3)	Kappa	.122	.080	4.291	.000

The McDonald Omega (ω) coefficient was used for the reliability analysis of the scale. Omega is shown by many researchers as a more reasonable indicator of internal consistency when compared to both alpha and other alternatives (Dunn, Baguley, & Brunsden, 2014). McDonald's ω coefficient is also called construct reliability (Nunnally & Bernstein, 1994), and can be obtained by confirmatory factor analysis or by hierarchical factor analysis (Yurdagül, 2006). On the other hand, ω is equal to or greater than α in all measurements (Bacon et al., 1995). One of the effective analysis programs for the calculation of Omega is the R program (Dunn, Baguley, Brunsden, 2014). In this study, the Jamovi package program was used for the analysis (https://www.jamovi.org).

FINDINGS

The computational thinking test was explained by the reasoning before the McDonald omega coefficient was chosen for reliability analysis. However, McDonald’s ω, together with Cronbach's alpha coefficient can be used to compare two reliability values. The Cronbach's alpha coefficient and the McDonald's ω coefficient for the internal consistency analysis of the developed test were 0.772 and 0.779, respectively. Data on CTT's item statistics are given in Table 4.

Table 4. Item Reliability Statistics

	Mean	Sd	item-test correlation	if item dropped McDonald's ω
1	7.836	2.91	0.472	0.760
2	0.591	2.00	0.103	0.795
3	0.609	1.27	0.420	0.764
4	7.473	3.40	0.598	0.742

	Mean	Sd	item-test correlation	if item dropped McDonald's ω
5	7.545	3.49	0.564	0.746
6	2.027	3.27	0.496	0.757
7	3.409	4.17	0.530	0.749
8	0.982	2.35	0.334	0.772
9	0.345	1.34	0.369	0.770
10	2.591	3.35	0.549	0.746

Table 4 demonstrates that, some items in the test, such as the item 2, are answered by students at a very low level, and therefore their correlation value with the average of all the tests is low. For this reason item 2 was deleted from the test and reliability analyses were repeated. Reliability statistics when item 2 dropped are shown in Table 5.

Table 5. Scale Reliability Statistics when item 2 dropped

	Mean	Sd	Cronbach's α	McDonald's ω
CTT	3.65	1.81	0.785	0.795

According to Table 5, the internal consistency of CTT was found to be sufficient (α=0,78, ω=0,79) and it can be considered as good reliability (Nunnally & Bernstein, 1994). The correlation value between the two halves of the test was calculated to obtain more evidence of the internal consistency of the test and the statistical values are given in Table 6.

Table 6. Guttman Split-Half and Spearman-Brown statistics

Reliability Statistics			
Cronbach's Alpha	Part 1	Value	,712
		N of Items	5[a]
	Part 2	Value	,574
		N of Items	4[b]
	Total N of Items		9
Correlation Between Forms			,616
Spearman-Brown Coefficient	Equal Length		,762
	Unequal Length		,764
Guttman Split-Half Coefficient			,748

It is seen in Table 6 that the internal consistency of the test according to this correlation value is at the Guttman Split-Half value of 0,748 and the two halves of the test is at the local level for the subsequent applications.

DISCUSSION

Computational thinking, is becoming an issue of great interest, and discussed quite often in the educational organizations in recent years. Many countries, such as Turkey and Korea, have begun to incorporate computational thinking skills in the software course curriculum (MEB, 2016). On the other hand, the industry-led 4.0, which was led by Germany, has also increased the need for individuals to have advanced thinking skills. However, since the research findings are new, discussions are still ongoing about how these skills can be measured. Barr, Harrison, and Leslie (2011) state that integration of IT into educational programs could help students improve their skills of using computer software, solve their insecurities against complex problems of nature, as well as improving their emotional skills such as working and communicating with others. Kalelioglu, Gülbahar and Kukul (2016) suggest that researchers should focus on the teaching and evaluation dimensions by developing scales and tasks that assess IT skills or by focusing on decisions that have a direct impact on them. Related with this point, the aim of this study was to develop a valid and reliable measurement tool that would measure the computational thinking skills of middle school eighth grade students.

For this reason, the aim of this research is to develop a valid and reliable Computational Thinking Test (CTT) that can be used at the eighth grade level. The computational thinking skills defined in the literature were identified and then the problematic situations which could measure these skills were established. CTT included 10 problem cases which were constructed in connection with the subject areas of science and mathematics education programs. According to analyses item two dropped from test and internal consistency of CTT was found as 0,79 (α=0,78, ω=0,79). This score may reveal as inefficient, but when the literature is examined, it is seen that the reliability coefficient is approximately at these levels in the tests designed to measure the information processing skills. For example, the Cronbach Alpha reliability coefficient of the scale which was adapted by Korkmaz, Çakır, and Özden (2016) is .809. Additionally Román-González, Pérez-González, & Jiménez-Fernández (2017) reported their alpha reliability coefficient of their test as 0.793. This is a situation that needs to be considered. Jun, Han, Kim, and Lee (2014) pointed out that the reason for the low scores of the students from the subscales of the computational thinking skill test in their study was related to the deficiencies in the curriculum and that new strategies should be developed in the field of education in order to improve the skills of the students. A similar result was obtained within the scope of this study. Although the problematic situations prepared based on science and mathematics education programs, it has been seen that the students have difficulty in some problem situations. One of the main reasons for this situation is that the content and activities that improve the ability of informatics thinking are not given as much as in the curriculum and that the students are foreign to such problem situations. Similarly, Kalelioğlu & Gülbahar (2015) reported that, the low student success rates in Turkey in this area may be due to the different approaches in the questions, and the variations in (or absence of) students' background knowledge. Taking into consideration the problems encountered in similar applications in the literature, it can be said that the test developed in this study is sufficient to measure students' computational thinking skills, and that the results obtained here will provide an important basis for future research.

REFERENCES

Angeli, C., Voogt, J., Fluck, A., Webb, M., Cox, M., Malyn-Smith, J., & Zagami, J. (2016). A K-6 Computational Thinking Curriculum Framework: Implications for Teacher Knowledge. Educational Technology & Society, 19 (3), 47–57.

Atılgan, H., Kan, A. ve Doğan, N. (2007). Eğitimde Ölçme ve Değerlendirme, Geliştirilmiş İkinci Baskı. Anı Yayıncılık. Ankara.

Atmatzidou, S. & Demetriadis, S. (2016). Advancing students' computational thinking skills through educational robotics: A study on age and gender relevant differences. *Robotics and Autonomous Systems*, *75*, 661-670.

Barr, D; Harrison, J; Conery, L. (2011). Computational Thinking: A Digital Age Skill for Everyone. Learning & Leading with Technology, v38 n6 p20-23 Mar-Apr 2011

Brettel, M., Friederichsen, N., Keller, M., & Rosenberg, M. (2014). How virtualization, decentralization and network building change the manufacturing landscape: An industry 4.0 perspective. *International Journal of Mechanical, Industrial Science and Engineering*, *8*(1), 37-44.

Csernoch, M., Biró, P., Máth, J. & Abari, K. (2015) Testing Algorithmic Skills in Traditional and Non-Traditional Programming Environments. *Informatics in Education*, 14(2), 175–197, DOI: 10.15388/infedu.2015.11

Dunn, T. J., Baguley, T., & Brunsden, V. (2014). From alpha to omega: A practical solution to the pervasive problem of internal consistency estimation. *British Journal of Psychology*, *105*(3), 399-412.

Fields, D. A., Searle, K. A., Kafai, Y. B., & Min, H. S. (2012). Debuggems to assess student learning in e-textiles. In *Proceedings of the 43rd SIGCSE Technical Symposium on Computer Science Education*. New York, NY: ACM Press.

Grover, S. & Pea, R. (2013). Computational Thinking in K–12: A Review of the State of the Field, Educational Researcher, 42(1), pp. 38–43 DOI: 10.3102/0013189X12463051

Han Koh, K., Basawapatna, A., Bennett V., & Repenning, A. (2010). Towards the automatic ecognition of computational thinking for adaptive visual language learning. In Proceedings of the 2010 Conference on Visual Languages and Human Centric Computing (VL/HCC 2010) (pp. 59–66). Madrid, Spain: IEEE Computer.

Israel, M., Pearson, J. N., Tapia, T., Wherfel, Q. M. & Reese, G. (2015). Supporting all learners in school-wide computational thinking: A cross-case qualitative analysis. Computers & Education 82, 263-279.

Jun, S., Han, S. Kim, H. & Lee, W. (2014). Assessing the computational literacy of elementary students on a national level in Korea Educational Assessment, Evaluation and Accountability, 26 (4), 319–332. DOI: 10.1007/s11092-013-9185-7

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

Kalelioğlu, F., & Gülbahar, Y. (2015). Bilge Kunduz: Uluslararası Enformatik Yarışması Pilot Uygulama Sonuçları. 9th International Computer & Instructional Technologies Symposium. Afyonkarahisar, Turkey, May 20-22, 2015.

Kim, B., Kim, T & Kim, J (2013). Paper-And-Pencil Programming Strategy toward Computational Thinking For Non-Majors: Design Your Solution. *J. Educational Computing Research*, Vol. 49(4) 437-459.

Korkmaz, Ö., Çakır, R., & Özden, M. Y. (2016). Bilgisayarca Düşünme Beceri Düzeyleri Ölçeğinin (BDBD) Ortaokul Düzeyine Uyarlanması. *Gazi Eğitim Bilimleri Dergisi*, *1*(2), 143-162

Korkmaz, Ö., Çakır, R., Özden, M. Y., Oluk, A., & Sarıoğlu, S. (2015). Bireylerin Bilgisayarca Düşünme Becerilerinin Farklı Değişkenler Açısından İncelenmesi. *Ondokuz Mayıs Üniversitesi Eğitim Fakültesi Dergisi*, *34*(2), 68-87.

MEB, 2016. Bilgisyar bilimi dersi öğretim programı, kur 1 – kur 2, MEB, Ankara.

Marshall, K. S. (2011). Assessing Computational Thinking Patterns. Presented at AERA 2011 Annual Meeting, New Orleans, LA

Nunnally, J. C., & Bernstein, I. H. (1994). Psychometric theory (3rd ed.). New York: McGraw-Hill.

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

Sanford, J. F., & Naidu, J. T. (2016). Computational thinking concepts for grade school. *Contemporary Issues in Education Research (Online), 9*(1), 23.

Swaid, S. I. (2015). Bringing computational thinking to STEM education. *Procedia Manufacturing*, *3*, 3657-3662.

Webb, D.C., 2010 Troubleshooting assessment: an authentic problem solving activity for it education, Procedia - Social and Behavioral Sciences, Volume 9, World Conference on Learning, Teaching and Administration Papers, pp. 903-907.

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L. & Wilensky, U. (2016) Defining Computational Thinking for Mathematics and Science Classrooms. *J Sci Educ Technol*, 25: 127–147. DOI 10.1007/s10956-015-9581-5

Werner, L., Denner, J., Campe, S., & Kawamoto, D. C. (2012). The Fairy performance assessment: Measuring computational thinking in middle school. In Proceedings of the 43rd ACM Technical Symposium on Computer Science Education (SIGCSE ’12), 215-220. New York,NY: ACM.

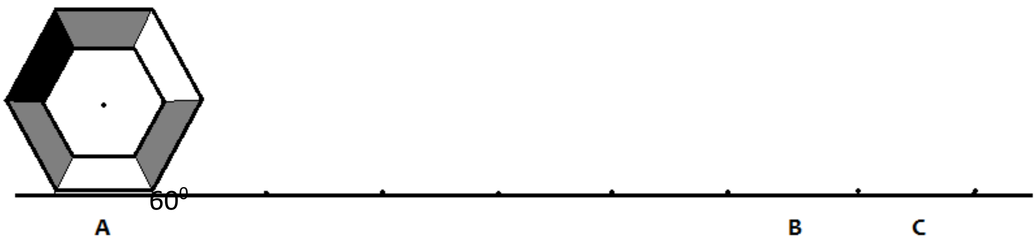
Wing, J. M. (2008). Computational thinking and thinking about computing. *Phil. Trans. R. Soc*. 366, 3717–3725

Yadav, A., Hong, H. & Stephenson, C. (2016). Computational Thinking for All: Pedagogical Approaches to Embedding 21st Century Problem Solving in K-12 Classrooms. TechTrends, 1-14 DOI 10.1007/s11528-016-0087-7

Yurdagül, H. (2006). The comparison of reliability coefficients in parallel, tau-equivalent, and congeneric measurements. Ankara University, Journal of Faculty of Educational Sciences, 39(1)

APPENDIX

Problem 1



The hexagonal object consisting of black, gray and white colors can be taken by turning it clockwise 60 degrees around its axis and shifting it on the base. In this process, the sign of " "refers rotation 60 degree around the axis, and the sign of the " " refers slip on the base of hexagonal object. Please answer the question based on this information.

a) What is the base color of the object when it reaches the "B" range without being slipped?

....

b) The object is moved downwards in this pattern;

What is the base color of the object in the "B" range as a result of this pattern?

....

c) When the object reaches the "C" range, the following application is made so that the base color can be black;

However, as a result of this application, it has been determined that the base color is not black. Find and correct this error and plot it in the space below.

....