# Complementary Tools for Computational Thinking Assessment

Marcos ROMÁN-GONZÁLEZ[1*], Jesús MORENO-LEÓN[2], Gregorio ROBLES[3]
[1] Universidad Nacional de Educación a Distancia, Spain
[2] Programamos.es & Universidad Rey Juan Carlos, Spain
[3] Universidad Rey Juan Carlos, Spain
[*]mroman@edu.uned.es, jesus.moreno@programamos.es, grex@gsyc.urjc.es

## ABSTRACT

Computational thinking (CT) is emerging as a key set of problem-solving skills that must be developed by the new generations of digital learners. However, there is still a lack of consensus on a formal CT definition, on how CT should be integrated in educational settings, and specially on how CT can be properly assessed. The latter is an extremely relevant and urgent topic because without reliable and valid assessment tools, CT might lose its potential of making its way into educational curricula. In response, this paper is aimed at presenting the convergent validity of one of the major recent attempts to assess CT from a *summative-aptitudinal* perspective: the Computational Thinking Test (CTt). The convergent validity of the CTt is studied in middle school Spanish samples with respect to other two CT assessment tools, which are coming from different perspectives: the Bebras Tasks, built from a *skill-transfer* approach; and Dr. Scratch, an automated tool designed from a *formative-iterative* approach. Our results show statistically significant, positive and moderately intense, correlations between the CTt and a selected set of Bebras Tasks ($r$=0.52); and between the CTt and Dr. Scratch (predictive value $r$=0.44; concurrent value $r$=0.53). These results support the statement that CTt is *partially convergent* with Bebras Tasks and with Dr. Scratch. Finally, we discuss if these three tools are complementary and may be combined in middle school.

## KEYWORDS

Computational thinking assessment, Computational Thinking Test, Dr. Scratch, Bebras Tasks, middle school.

## 1. INTRODUCTION

Computational thinking (CT) is considered in many countries as a key set of problem-solving skills that must be acquired and developed by today's generation of learners (Bocconi et al., 2016). However, there is still a lack of consensus on a formal CT definition (Kalelioglu, Gülbahar, & Kukul, 2016), on how CT should be integrated in educational settings (Lye & Koh, 2014), and especially on how CT can be properly assessed (Grover, 2015; Grover & Pea, 2013). Regarding to the latter, even though computing is being included into K-12 schools all around the world, the issue of assessing student's CT remains a thorny one (Grover, Cooper, & Pea, 2014). Hence, CT assessment is an extremely relevant and urgent topic to address, because "without attention to assessment, CT can have little hope of making its way successfully into any K-12 curriculum", and consequently "measures that would enable educators to assess what the child has learned need to be validated" (Grover & Pea, 2013, p. 41).

Moreover, from a psychometric approach, CT is still a poorly defined psychological construct as its nomological network has not been completely established; that is, the correlations between CT and other psychological constructs have not been completely reported by the scientific community yet (Román-González, Pérez-González, & Jiménez-Fernández, 2016). Furthermore, there is still a large gap of tests relating to CT that have undergone a comprehensive psychometric validation process (Mühling, Ruf, & Hubwieser, 2015). As Buffum et al. (2015) say: "developing (standardized) assessments of student learning is an urgent area of need for the relatively young computer science education community" (Buffum et al., 2015, p. 622)

In order to shed some light on this issue, one of the major attempts to develop a solid psychometric tool for CT assessment is the Computational Thinking Test (CTt) (Román-González, 2015). This is a multiple-choice test that has demonstrated to be valid and reliable ($\alpha$=0.80; $r_{xx}$=0.70) in middle school subjects, and which has contributed to the nomological network of CT in regard to other cognitive (Román-González, Pérez-González, & Jiménez-Fernández, 2016) and non-cognitive (Román-González, Pérez-González, Moreno-León, & Robles, 2016) key psychological constructs. Continuing this research line, now we investigate the convergent validity of the CTt, that is, the correlations between this test and other tools aimed at assessing CT. Thus, our general research question is:

*RQ $_{(general)}$: What is the convergent validity of the CTt?*

### *1.1. Computational thinking assessment tools*

Focusing on K-12 education, especially in middle school and without being exhaustive, we find several CT assessment tools developed from different perspectives:

***CT Summative tools.*** We can differentiate between: a) Aptitudinal tests such as the aforementioned *Computational Thinking Test* (which is further described in 2.1.), the *Test for Measuring Basic Programming Abilities* (Mühling et al., 2015), or the *Commutative Assessment Test* (Weintrop & Wilensky, 2015). And b) Content-knowledge assessment tools such as the summative tools of Meerbaum-Salant et al. (2013) in the Scratch context, or those used for measuring the students' understanding of computational concepts after introducing a new computing curriculum (e.g., in Israel, Zur-Bargury, Pârv, & Lanzberg, 2013).

***CT Formative-iterative tools.*** They provide feedback, usually in an automatic way, for learners to improve their CT skills. These tools are specifically designed for a particular programming environment. Thus, we find *Dr. Scratch* (Moreno-León & Robles, 2015) or *Ninja Code Village* (Ota, Morimoto, & Kato, 2016) for Scratch; the ongoing work of Grover et al. (2016) for Blockly; or the *Computational Thinking Patterns CTP-Graph* (Koh, Basawapatna, Bennett, & Repenning, 2010) for AgentSheets.

***CT Skill-Transfer tools.*** They are aimed at assessing the students' transfer of their CT skills to different types of problems: for example, the *Bebras Tasks* (Dagiene & Futschek, 2008) focused on measuring transfer to 'real-life' problems; or the *CTP-Quiz* (Basawapatna, Koh, Repenning, Webb, & Marshall, 2011), which evaluates the transfer of CT to the context of scientific simulations.

***CT Perceptions-Attitudes scales,*** such as the *Computational Thinking Scales* (CTS) (Korkmaz, Çakir, & Özden, 2017), which uses five-point Likert scales and has been recently validated with Turkish students.

***CT Vocabulary assessments.*** They are aimed at measuring elements and dimensions of CT verbally expressed by children (i.e. 'computational thinking language'; e.g. Grover, 2011).

Using only one type from the aforementioned assessment tools can lead to misunderstand the development of CT skills by students. In this sense, Brennan and Resnick (2012) have stated that looking at student-created programs alone could provide an inaccurate sense of students' computational competencies, and they underscore the need for multiple means of assessment. Therefore, as it has been pointed out by relevant researchers (Grover, 2015; Grover et al., 2014), in order to reach a total and comprehensive understanding of the CT of our students, different types of complementary assessments tools must be systematically combined (i.e. also called "systems of assessments"). Following this idea, our paper is specifically aimed at studying the convergent validity of the CTt with respect to other assessment tools, which are coming from different perspectives. Thus, our specific research questions are:

*RQ $_{(specific\text{-}1)}$: What is the convergent validity between CTt and Bebras Tasks? RQ $_{(specific\text{-}2)}$: What is the convergent validity between CTt and Dr. Scratch?*

Although the three instruments involved in our research are aimed at assessing the same construct (i.e. CT), as they approach the measurement from different perspectives, a *total convergence* ($r$>0.7) is not expected among them, but a *partial* one (0.4<$r$<0.7) (Carlson & Herdman, 2012). Answering the aforementioned questions may contribute to develop a comprehensive "system of assessment" for CT in middle school settings.

## 2. BACKGROUND

### *2.1. Computational Thinking Test (CTt)*

The Computational Thinking Test[1] (CTt) is a multiple-choice instrument composed by 28 items, which are administered on-line (via non-mobile or mobile electronic devices) in a maximum time of 45 minutes. Each item of the CTt is presented either in a 'maze' or in a 'canvas' interface; and is designed according to the following three dimensions (Román-González, 2015; Román-González, Pérez-González, & Jiménez-Fernández, 2016):

- **Computational concept addressed:** each item addresses one or more of the following seven computational concepts, ordered in increasing difficulty: Basic directions and sequences; Loops–repeat times; Loops–repeat until; If–simple conditional; If/else–complex conditional; While conditional; Simple functions. These 'computational concepts' are progressively nested along the test, and are aligned with the CSTA Computer Science Standards for the 7th and 8th grade (Seehorn et al., 2011).
- **Style of answers:** in each item, responses are presented in any of these two styles: 'visual arrows' or 'visual blocks'.
- **Required task:** depending on which cognitive task is required for solving the item: 'sequencing' ≈ stating in an orderly manner a set of commands, 'completion' of an incomplete set of commands, or 'debugging' an incorrect set of commands.

We show an example of a CTt item translated into English in Figure 1, with its specifications detailed below.

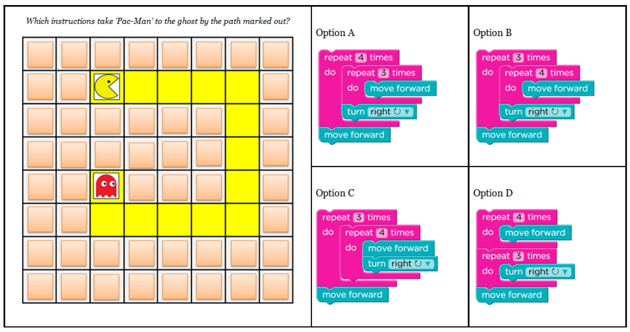


*Figure 1*. CTt, item nº 8 ('maze'): loops-repeat times (nested); visual blocks; sequencing.

[1] Sample copy available at: https://goo.gl/GqD6Wt.

### 2.2. *Bebras Tasks*

The Bebras Tasks are a set of activities designed within the context of the *Bebras International Contest*[2], a competition born in Lithuania in 2003 which aims to promote the interest and excellence of primary and secondary students around the world in the field of Computer Science from a CT perspective (Dagiene & Futschek, 2008; Dagiene & Stupuriene, 2015). Each year, the contest launches a set of Bebras Tasks, whose overall approach is the resolution of 'real-life' and significant problems, through the transfer and projection of the students' CT. These Bebras Tasks are independent from any particular software or hardware, and can be administered to individuals without any prior programming experience. For all these features, the Bebras Tasks have been pointed out to more than likely be an embryo for a future PISA (Programme for International Student Assessment) test in the field of Computer Science (Hubwieser & Mühling, 2014). As an example, one of the Bebras Tasks used in our research is shown in Figure 2.

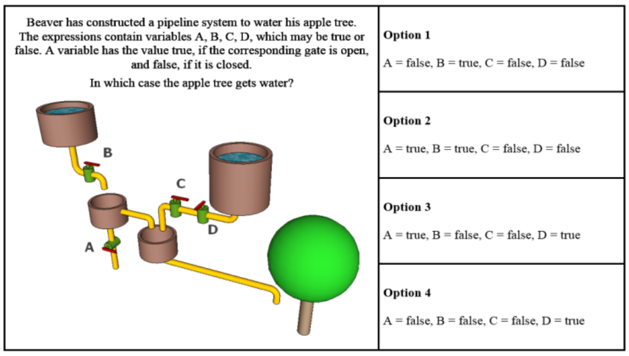


*Figure 2*. Example of a Bebras Task (*'Water Supply'*).

### 2.3. *Dr. Scratch*

Dr. Scratch[3] (Moreno-León & Robles, 2015) is a free and open source web application designed to analyze, in an automated way, projects programmed with Scratch. In addition, the tool provides feedback that middle school students can use to improve their programming and CT skills (Moreno-León, Robles, & Román-González, 2015). Therefore, Dr. Scratch is an automated tool for the formative assessment of Scratch projects.

As summarized in Table 1, the CT score that Dr. Scratch assigns to a project is based on the level of development of seven dimensions of the CT competence. These dimensions are statically evaluated by inspecting the source code of the analyzed project and given a punctuation from 0 to 3, resulting in a total evaluation ('mastery score') that ranges from 0 to 21 when all seven dimensions are aggregated.

Figure 3, which shows the source code of a Scratch project, can be used to illustrate the assessment of the tool. Dr. Scratch would assign 8 points of 'mastery score' to this project: 2 points for logical thinking, since it includes an 'if-else' statement; 2 points for user interactivity, as players interact with the sprite by using the mouse; 2 points for data representation, because the project makes use of a variable; 1 point for abstraction and problem decomposition, since there are two scripts in the project; and 1 point for flow control, because the programs are formed by a sequence of instructions with no loops. Parallelism and synchronization dimensions would be measured with 0 points.

*Table 1*. Dr. Scratch's score assignment.

| CT dimension | Competence Level: Basic (1 point) | Medium (2 points) | Proficient (3 points) |
|---|---|---|---|
| Abstraction and problem decomposition | More than one script | Use of custom blocks | Use of 'clones' (instances of sprites) |
| Logical thinking | If | If else | Logic operations |
| Synchronization | Wait | Message broadcast, stop all, stop program | Wait until, when backdrop changes, broadcast and wait |
| Parallelism | Two scripts on green flag | Two scripts on key pressed or sprite clicked | Two scripts on receive message, video/audio input, backdrop change |
| Flow control | Sequence of blocks | Repeat, forever | Repeat until |
| User interactivity | Green flag | Keyboard, mouse, ask and wait | Webcam, input sound |
| Data representation | Modifiers of object properties | Variables | Lists |

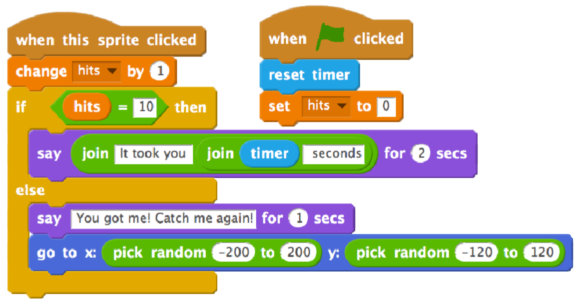


*Figure 3*. Source code of 'Catch me if you can 2'.
Available at https://scratch.mit.edu/projects/142454426/

Dr. Scratch is currently under validation process, although its convergent validity with respect to other traditional metrics of software complexity has been already reported (Moreno-León, Robles, & Román-González, 2016).

## 3. METHODOLOGY AND RESULTS

The convergent validity of the CTt with respect to Bebras Tasks and Dr. Scratch was investigated through two different correlational studies, with two independent samples.

[2] http://www.bebras.org/

[3] http://drscratch.org/

### *3.1. First study: CTt * Bebras Tasks*

Within the context of a broader pre-post evaluation of Code.org courses, the CTt and a selection of three Bebras Tasks were concurrently administered to a sample of *n*=179 Spanish middle school students (Table 2). This occurred only in pre-test condition, i.e. students without prior formal experience in programming and before starting with Code.org.

*Table 2.* Sample of the first study

| | 7th Grade | 8th Grade | Total |
|---|---|---|---|
| Boys | 88 | 15 | 103 |
| Girls | 60 | 16 | 76 |
| Total | 148 | 31 | 179 |

The three Bebras Tasks[4] were selected attending to following criteria: the activities were aimed to students in the range of 11-14 y/o, and focused in different aspects of CT. In Table 3, the correlations between the CTt score (which ranges from 0 to 28), the score in each of the Bebras Tasks (0 to 1), and the overall Bebras score for all of them (0 to 3) are shown. As the normality of the variables is not assured [p-value$_{(Zk\text{-}s)}$>0.05], non-parametric correlations are calculated (Spearman's *r*).

*Table 3*. Correlations CTt * Bebras Tasks (*n*=179)

| | Task #1: 'Water Supply' | Task #2: 'Fast Laundry' | Task #3: 'Abacus' | Whole Set of Tasks |
|---|---|---|---|---|
| CTt | .419** | .042 | .490** | .519** |

** p-value $_{(r)}$ < 0.01

As it can be seen, the CTt has a positive, moderate, and statistically significant correlation (*r*=0.52) with the whole set of Bebras Tasks (Figure 4); and with Tasks #1 ('Water Supply', related to logic-binary structures) and #3 ('Abacus', related to abstraction, decomposition and algorithmic thinking). No correlation is found between the CTt and Task #2 ('Fast Laundry', related to parallelism), which is consistent with the fact that CTt does not involve parallelism.

### *3.2. Second study: CTt * Dr. Scratch*

The context of this study is an 8-weeks coding course in the Scratch platform, following the *Creative Computing* (Brennan, Balch, & Chung, 2014) curriculum and involving three Spanish middle schools, with a total sample of *n*=71 students from the 8th Grade (33 boys and 38 girls).

Before starting with the course, the CTt was administered to the students in pre-test conditions (i.e. students without prior formal experience in programming). After the coding course, students took a post-test with the CTt and teachers selected the most advanced project of each student, which was analyzed with Dr. Scratch. These three measures offered us the possibility to analyze the convergent validity of the CTt and Dr. Scratch in predictive terms (CTt$_{pre\text{-}test}$*Dr. Scratch) and in concurrent terms (CTt$_{post\text{-}test}$*Dr. Scratch). As the normality of the variables is not assured either [p-value$_{(Zk\text{-}s)}$>0.05], non-parametric correlations (Spearman's *r*) are calculated again (Table 4).

*Table 4*. Correlations CTt * Dr. Scratch (*n*=71)

| | CTt Pre-test | CTt Post-test |
|---|---|---|
| Dr. Scratch ('mastery score') | .444** | .526** |

** p-value $_{(r)}$ < 0.01

As it can be seen, the CTt has a positive, moderate, and statistically significant correlation with Dr. Scratch, both in predictive (*r*=0.44) and concurrent terms (*r*=0.53, see Figure 5). As expected, the concurrent value is slightly higher because no time is intermediating among the tools.

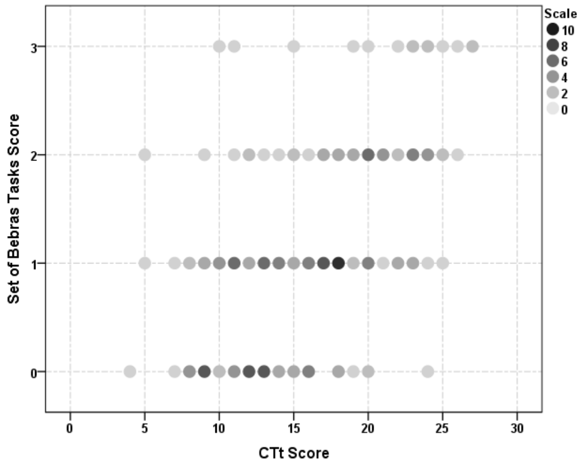


*Figure 4*. Scatterplot CTt * Set of Bebras Tasks.

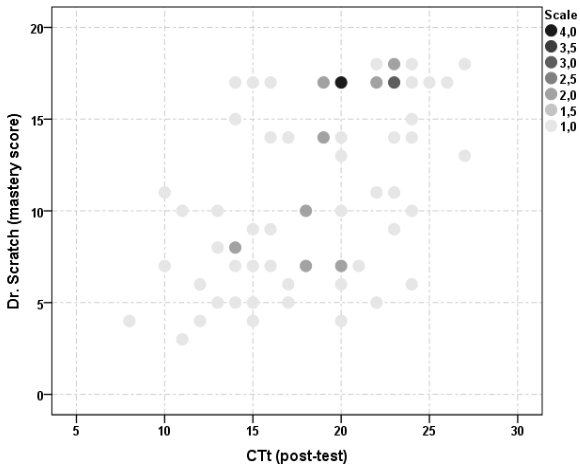


*Figure 5*. Scatterplot CTt $_{post\text{-}test}$*Dr. Scratch.

## 4. DISCUSSION AND CONCLUSIONS

Returning to our specific research questions, we have found that the CTt is *partially convergent* with the Bebras Tasks and with Dr. Scratch (0.4<*r*<0.7). As we expected, the convergence is not *total* (*r*>0.7) because, although the three tools are assessing the same psychological construct (i.e. CT), they do it from different perspectives:

[4] The Bebras Tasks used in our research, and their specifications, can be reviewed with more detail in: https://goo.gl/FXxgCz.

*summative-aptitudinal* (CTt), *skill-transfer* (Bebras Tasks), and *formative-iterative* (Dr. Scratch). On the one hand, these empirical findings imply that none of these tools should be used instead of any of the others, as the different scores are only moderately correlated (i.e. a measure from one of the tools cannot substitute completely the others); otherwise, the three tools might be combined in middle school contexts. On the other hand, from a theoretical point of view, the three tools seem to be complementary, as the weaknesses of the ones are the strengths of the others.

The CTt has some strengths such as: it can be collectively administered in pure pre-test conditions, so it can be used in massive screenings and early detection of students with high abilities (or special needs) for programming tasks; and it can be utilized for collecting quantitative data in pre-post evaluations of the efficacy of curricula aimed at fostering CT. However, it also has some obvious weakness: it provides a static and decontextualized assessment, and it is strongly focused on computational 'concepts' (Brennan & Resnick, 2012), ignoring 'practices' and 'perspectives'.

As a counterbalance of the previous weakness, the Bebras Tasks provides a naturalistic and significant assessment, which is contextualized in 'real-life' problems that can be used not only for measuring but also for teaching and learning CT. However, the psychometric properties of these tasks are still far of being demonstrated, and some of them are at risk of being too tangential to the core of CT.

Finally, Dr. Scratch complements the CTt as the former includes 'computational practices' (Brennan & Resnick, 2012) that the others do not, such as iterating, testing, remixing or modularizing. However, Dr. Scratch lacks the possibility of being used in pure pre-test conditions, as it is applied to Scratch projects after the student has learnt at least some coding for a certain time.

All of the above leads us to affirm the complementarity of the CTt, Bebras Tasks and Dr. Scratch in middle school settings; and the possibility to build up a "system of assessments" (Grover, 2015; Grover et al., 2014) with all of them. Furthermore, we find evidence to consider an analogous progression between the Bloom's (revised) taxonomy of cognitive processes (Krathwohl, 2002), and the three assessment tools considered along this paper (Figure 6).

## 5. LIMITATIONS AND FURTHER RESEARCH

Regarding the convergent validity of the CTt, another correlation value might have been found with Bebras Tasks if the researchers had selected a different set of them; also, another correlation value might have been found with Dr. Scratch if the teachers had selected a different set of projects. Further research should lead us to explore the convergent validity of the CTt with other assessment tools. For example, we are currently designing an investigation to study the convergence between the CTt and the Computational Thinking Scales (CTS) (Korkmaz et al., 2017), and another one that will study the convergence between Dr. Scratch and Ninja Code Village (Ota et al., 2016). As a major result of these future series of studies, it will be possible to depict a map with the convergence values between the main CT assessment tools all around the world, which ultimately would take CT to be well and seriously considered as a psychological construct.

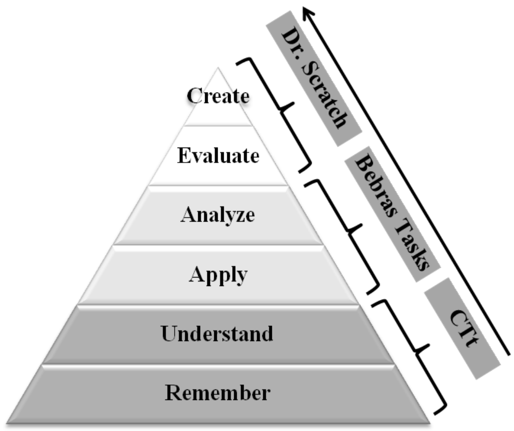


*Figure 6*. Bloom's taxonomy and CT assessment tools.

## 6. REFERENCES

Basawapatna, A., Koh, K. H., Repenning, A., Webb, D. C., & Marshall, K. S. (2011). Recognizing computational thinking patterns. In *Proceedings of the 42nd ACM technical symposium on Computer science education* (pp. 245–250).

Bocconi, S., Chioccariello, A., Dettori, G., Ferrari, A., Engelhardt, K., & others. (2016). *Developing Computational Thinking in Compulsory Education-Implications for policy and practice*.

Brennan, K., Balch, C., & Chung, M. (2014). Creative computing. *Harvard Graduate School of Education*.

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 annual meeting of the American Educational Research Association, Vancouv., Canada* (pp. 1–25).

Buffum, P. S., Lobene, E. V, Frankosky, M. H., ... , & Lester, J. C. (2015). A practical guide to developing and validating computer science knowledge assessments with application to middle school. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education* (pp. 622–627).

Carlson, K. D., & Herdman, A. O. (2012). Understanding the impact of convergent validity on research results. *Organizational Research Methods*, *15*(1), 17–32.

Dagiene, V., & Futschek, G. (2008). Bebras international contest on informatics and computer literacy: Criteria for good tasks. In *International Conference on Informatics in Secondary Schools-Evolution and Perspectives* (pp. 19–30).

Dagiene, V., & Stupuriene, G. (2015). Informatics education based on solving attractive tasks through a contest. *KEYCIT 2014: Key Competencies in Informatics and ICT*, *7*, 97.

Grover, S. (2011). Robotics and engineering for middle and high school students to develop computational thinking. In *annual meeting of the American Educational Research Association, New Orleans, LA*.

Grover, S. (2015). "Systems of Assessments" for Deeper Learning of Computational Thinking in K-12. In *Proceedings of the 2015 Annual Meeting of the American Educational Research Association* (pp. 15–20).

Grover, S., Bienkowski, M., Niekrasz, J., & Hauswirth, M. (2016). Assessing Problem-Solving Process At Scale. In *Proceedings of the Third (2016) ACM Conference on Learning@ Scale* (pp. 245–248).

Grover, S., Cooper, S., & Pea, R. (2014). Assessing computational learning in K-12. In *Proceedings of the 2014 conference on Innovation & technology in computer science education* (pp. 57–62).

Grover, S., & Pea, R. (2013). Computational Thinking in K--12 A Review of the State of the Field. *Educational Researcher*, *42*(1), 38–43.

Hubwieser, P., & Mühling, A. (2014). Playing PISA with bebras. In *Proceedings of the 9th Workshop in Primary and Secondary Computing Education* (pp. 128–129).

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.

Koh, K. H., Basawapatna, A., Bennett, V., & Repenning, A. (2010). Towards the automatic recognition of computational thinking for adaptive visual language learning. In *Visual Languages and Human-Centric Computing, 2010 IEEE Symposium on* (pp. 59–66).

Korkmaz, Ö., Çakir, R., & Özden, M. Y. (2017). A validity and reliability study of the Computational Thinking Scales (CTS). *Computers in Human Behavior*.

Krathwohl, D. R. (2002). A revision of Bloom's taxonomy: An overview. *Theory into Practice*, *41*(4), 212–218.

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

Meerbaum-Salant, O., Armoni, M., & Ben-Ari, M. (2013). Learning computer science concepts with scratch. *Computer Science Education*, *23*(3), 239–264.

Moreno-León, J., & Robles, G. (2015). Dr. Scratch: A web tool to automatically evaluate Scratch projects. In *Proceedings of the Workshop in Primary and Secondary Computing Education* (pp. 132–133).

Moreno-León, J., Robles, G., & Román-González, M. (2015). Dr. Scratch: automatic analysis of scratch projects to assess and foster computational thinking. *RED. Revista de Educación a Distancia*, *15*(46).

Moreno-León, J., Robles, G., & Román-González, M. (2016). Comparing computational thinking development assessment scores with software complexity metrics. In *Global Engineering Education Conference, 2016 IEEE* (pp. 1040–1045).

Mühling, A., Ruf, A., & Hubwieser, P. (2015). Design and first results of a psychometric test for measuring basic programming abilities. In *Proceedings of the Workshop in Primary and Secondary Computing Education* (pp. 2–10).

Ota, G., Morimoto, Y., & Kato, H. (2016). Ninja code village for scratch: Function samples/function analyser and automatic assessment of computational thinking concepts. In *Visual Languages and Human-Centric Computing (VL/HCC), 2016 IEEE Symposium on* (pp. 238–239).

Román-González, M. (2015). Computational Thinking Test: Design Guidelines and Content Validation. In *Proceedings of the 7th Annual International Conference on Education and New Learning Technologies (EDULEARN 2015)* (pp. 2436–2444).

Román-González, M., Pérez-González, J.-C., & Jiménez-Fernández, C. (2016). Which cognitive abilities underlie computational thinking? Criterion validity of the Computational Thinking Test. *Computers in Human Behavior*.

Román-González, M., Pérez-González, J.-C., Moreno-León, J., & Robles, G. (2016). Does Computational Thinking Correlate with Personality?: The Non-cognitive Side of Computational Thinking. In *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality* (pp. 51–58).

Seehorn, D., Carey, S., Fuschetto, B., Lee, I., Moix, D., O'Grady-Cunniff, D., … Verno, A. (2011). CSTA K--12 Computer Science Standards: Revised 2011.

Weintrop, D., & Wilensky, U. (2015). Using Commutative Assessments to Compare Conceptual Understanding in Blocks-based and Text-based Programs. In *ICER* (Vol. 15, pp. 101–110).

Zur-Bargury, I., Pârv, B., & Lanzberg, D. (2013). A nationwide exam as a tool for improving a new curriculum. In *Proceedings of the 18th ACM conference on Innovation and technology in computer science education* (pp. 267–272).