Exploring Computational Thinking Assessment in Introductory Programming Courses

Ana Liz Souto O. Araujo*†, Jucelio S. Santos*, Wilkerson L. Andrade*, Dalton D. Serey Guerrero*, Valentina Dagienė‡
*Federal University of Campina Grande, Paraíba, Brazil
†Federal University of Paraíba, Paraíba, Brazil
‡Vilnius University, Vilnius, Lithuania
analiz@dcx.ufpb.br, jucelio@coping.ufcg.edu.br, {wilkerson, dalton}@computacao.ufcg.edu.br
valentina.dagiene@mii.vu.lt

***Abstract*—Several instruments have been used for assessing Computational Thinking (CT) abilities. In this exploratory and preliminary study, we investigate how appropriate the Bebras challenge is as an instrument to assess and measure CT abilities. Bebras is an international challenge whose goal is to promote Computer Science and CT. The test can be answered without any prior knowledge on computer science. Our broad research question is whether we can evolve Bebras into a full fledged instrument to assess and measure CT abilities. In this paper, we instantiate a few more specific research questions: Is Bebras performance a good predictor of success for students within programming courses? Is there any correlation between Bebras performance and students' grades? Do students improve their performance in Bebras tests when exposed to the contents of a programming course? Our dataset consists of the grades of 138 students who attended introductory programming courses at two Brazilian universities and their performance in two simulated Bebras tests. The first test was applied at the beginning of the term and therefore before any exposition to programming classes. The second one was applied at the end of the term. The results suggest that the performance on Bebras is only moderately correlated to the student grades. We conclude that it is not very likely that CT measures can be derived from the Bebras test as it is currently designed. While further research is needed on how we can leverage the Bebras effort to extend it into a CT assessment instrument, we performed a preliminary study on the use of Item Response Theory (IRT) as a means to improve the selection of questions and the design of the test. We expect the results of this research can contribute both to the development and discussion on CT assessment as well as to the Bebras effort to educate CT.**

I. Introduction

Computational Thinking (CT) involves solving problems, designing systems, and understanding human behavior by drawing on the fundamental concepts of Computer Science (CS) [1]. CT encompasses all mental tools that reflect and support the breadth of the Computer Science field. Thus, CT is more about the cognitive abilities used to solve problems within CS than about technical skills.

To develop children's CT abilities we must be able to assess and measure them. Besides improving our understanding of the cognitive abilities themselves, developing ways to assess and measure CT abilities would allow us to compare and relate them to cognitive abilities required in related areas, such as math, science and engineering. Assessing CT abilities, however, is a challenge because it means measuring latent variables that cannot be directly observed.

Tests and coding problems are the most popular artifacts used to assess CT abilities [2]. Tests are sets of objective questions designed to measure the respondents' knowledge. However, there is no validated and widely accepted approach to create tests to assess CT abilities. As evidence, different studies use significantly different tests to perform the assessment. Using coding problems and programming projects to assess CT abilities does not make things better [2]. In this approach, CT abilities are assessed based on actual programming skills of the subjects. In most cases computer code must be written by the subjects. The assessment itself usually consists in verifying whether the code produced by the subject contains certain structural elements from within a predefined checklist. While these approaches are useful for measuring students' problem-solving skills, they require the subjects to have some actual knowledge on computer programming. Being able to assess the fundamental CT abilities without resorting to concrete programming skills would be useful. Thus, to make the assessment independent of previous training in computer science technology, tests seem to be a better way.

Developing appropriate tests to measure students' cognitive abilities, however, is not an easy endeavor. In our opinion, it requires knowledge from widely different areas of expertise. In particular, we believe and defend that it should leverage on the knowledge developed by psychometrics. According to psychometrics, appropriate testing requires a rigorous validation process before it can be effectively used as a measuring instrument [3].

Bebras is an international effort that builds on the idea that teaching computer programming directly is not mandatory to stimulate CT. Bebras is an international challenge whose goal is to promote Computer Science and CT among primary and secondary school students, and also for the general public [4]. The main part of the Bebras challenge is a contest run annually one or two times for six different age groups. The contest is a test composed of multiple choice questions/short answer as well as interactive tasks related to problem-solving that does not depend on programming. Although the test can be answered without prior knowledge about computer

978-1-5090-5920-1/17/$31.00 ©2017 IEEE

programming and computer science in general, all questions are indirectly related to fundamental computational concepts. Bebras was created in 2004 (bebras.org). In 2016, more than 1.5 million students from 38 different countries participated in the contest.

Bebras, however, was proposed as a means to promote and stimulate CT among students. It was not designed to be a measurement instrument. A few studies have suggested that Bebras could be used as an assessment instrument to assess CT abilities. They were motivated because many schools already prepared their students to take the test because it is an external and international competition [5]. Most popular themes in these studies are gender issues, difference among age and countries [6]–[9]. A few studies also explore the relation between CT and programming abilities [5], [10].

In this study, we investigate how appropriate Bebras can be as an instrument to assess CT abilities. Our study has been developed within the context of an introductory programming course. We explore the relation between students performance in simulated multiple choice Bebras tests and grades in an introductory programming course. Our motivating research questions are: Is Bebras performance a good predictor of success for students within programming courses? Is there any correlation between Bebras performance and students' grades? Do students improve their performance in Bebras tests when exposed to the contents of a programming course?

Our dataset consists of the grades of 138 students who attended introductory programming courses at two Brazilian universities and their performance in two simulated multiple choice Bebras tests. The first test was applied at the beginning of the term and therefore before any exposition to programming classes. The second Bebras test was applied at the end of the term. The results suggest that the performance on Bebras tasks is only moderately correlated to the student grades. We conclude that it is not very likely that CT measures can be derived from the Bebras test as it is currently designed. While further research is needed on how we can leverage the Bebras effort to extend it into a CT assessment instrument, we performed a preliminary study on the use of Item Response Theory (IRT) as a means to improve the selection of questions and the design of the test. We expect the results of this research can contribute to both the development and discussion on CT assessment as well as the Bebras effort to disseminate CT.

The remainder of this paper is organized as follows: Section II introduces related work about assessing CT; Section III presents the Bebras challenge; Section IV details instruments, participants, and procedures of the study; Section V exposes the results; Section VI discusses the obtained results and the validity threats of our study; Section VII describes a preliminary study of the applicability of Item Response Theory methods to the Bebras; finally, Section VIII presents the concluding remarks and future works.

II. Assessing Computational Thinking

Programming courses with visual programming languages are the most popular approach not only to stimulate but also to assess CT [2]. Scalable Game Design is an ongoing project to develop and assess CT in the context of games and scientific simulations [11]. AgentSheets and AgentCubes are the tools used for developing games and express CT abilities. In the context of the project, Computational Thinking Patterns are abstract programming patterns that enable agent interactions in games and science simulations [12]. The assessment is based on an automatic suite to detected common design patterns in the students' project.

Frameworks for assessing CT based on Scratch are also widely used. Brennan and Resnick proposed a remarkable framework for assessing computational thinking practices with Scratch [13]. The framework focuses on three approaches: *(i)* artifact-based interviews involving several categories of questions (defining Scratch, providing feedback, solving problems, and developing projects); *(ii)* design scenarios where learners encounter a series of projects and engage with them from four perspectives: critiquing, extending, debugging, and remixing; and *(iii)* learner documentation where participants are engaged in developing reflective traces of their learning with a journal or comments in code. This framework is based on a qualitative assessment approach.

The Progression of Early Computational Thinking Model is another framework for understanding and assessing CT in the primary grades [14]. The model synthesizes measurable evidence from student's Scratch project, such as coding design patterns, which is then mapped onto computational thinking concepts. This model assumes that every student has a proficiency level of CT that manifests itself in the student's ability to design and code specific tasks.

Different from frameworks and specific tools, other studies use programming tests for measuring CT. Bargury *et al.* reported a test to assess the CS curriculum of Israel for junior high, which focuses on developing CT [15]. The test consists of 11 questions about language proficiency questions, variable questions, and questions that require algorithmic thinking in Scratch. The evaluation focuses on the student's ability to cope with algorithmic thinking based on conditional and loop structures.

Another two studies are based on questions of the same test mentioned above to assess middle school student learning of programming and CT in Scratch [16], [17]. In both studies, the authors created pre and post CT tests based on Scratch for students enrolled in an introductory CS course for middle school. In [16], the results showed that the CT pre-test scores are strong predictors of performance on the CT post-test. In [17], the results indicated that all students have higher averages on post-tests than pre-tests. Both studies reported a strong correlation between CT and programming.

Despite the studies above reported the assessment of CT based on programming activities, other initiatives expose investigation of measuring CT without programming knowledge. Gouws *et al.* has designed and administered a test for CT ability for introductory computer science students [18]. The content of the test is structured to test general problem-solving skills and not programming. They contrasted the results of

the test with the Computer Science 101 (CSC 101) grades. The results indicated that students who performed well in the assessment have a favorable pass rate for their class tests. Marais and Bradshaw [19] have designed a pre and post test based on the test proposed by [18]. They have explored the develop of problem-solving skills in first-year computer science students. The findings suggest that students achieve problem-solving skills after completing first-year computer science course and the skills are both innate in some students and acquired in others.

Pilot studies have explored Bebras challenge as an instrument for assessing CT. Duncan and Bell analyzed three CS curricula for primary schools and assessed CT abilities with tests, survey and class observation [5]. Concerned about the tests, they used Bebras challenge, binary numbers activities, and programming. The results showed some correlation between the Bebras results and the score for binary numbers activities, and between programming ability and binary numbers activities. By contrast, they found a weak correlation between Bebras performance and the programming test. The author argued that the Bebras test might be used as a test of CT skills, but more investigation is required to establish this.

Dolgopolovas *et al.* investigated CT and programming among novice software engineering students using Bebras [10]. They administered Bebras at a structured programming course. They preselected Bebras tasks with the focus on algorithmic thinking and according to international expert difficult evaluations. The Rasch Model estimation for test items of Bebras had shown the possible validity of the test as an instrument to assess CT abilities. On the other hand, the results did not present any correlations between CT and latent abilities measured at structured programming course.

III. Bebras

Bebras challenge is an international initiative whose goal is to promote CS and CT for primary and secondary school students (bebras.org). The challenge aims at attracting student to informatics and stimulating CT. Although the test can be answered without prior knowledge about computing, all questions are related to computational concepts. The Bebras' tasks are usually related to one or more of the following categories: *(i)* information comprehension; *(ii)* algorithmic thinking; *(iii)* using computer systems; *(iv)* structures, patterns and arrangements; *(v)* puzzles; *(vi)* social, ethical, cultural, international, and legal issues [4]. In 2017, a new two-dimensional categorization system is proposed [20]. This new two-dimensional categorization system incorporates both CT skills and informatics concepts in the classification of tasks. The CT skills are Abstraction, Algorithmic thinking, Decomposition, Evaluation, Generalization. The informatics concepts are *(i)* Algorithms and programming, *(ii)* Data, data structures and representations, *(iii)* Computer processes and hardware, *(iv)* Communication and networking, and *(v)* Interactions, systems and society.

Bebras has been applied by age group. The name and grades of the group can vary among countries, but in most of the cases are Pre-Primary (grades 1-2), Little Beavers (grades 3- 4), Benjamin (grades 5-6), Cadet (grades 7-8), Junior (grades 9-10), Senior (grades 11-12). Frequently, the contest is performed in schools using computers but also can be performed with paper and pencil. The participants are supervised by teachers who may integrate the contest into their teaching activities [4]. Pupils should solve 15 to 18 tasks within 40 to 55 minutes.

TABLE I
Scores of the original Bebras challenge

Difficulty	Easy	Medium	Hard
Correct answer	6	9	12
No answer	0	0	0
Incorrect answer	0	-2	-4

Bebras has been arranged annually since 2004. Once a year, researchers from all the countries involved meet in a workshop for the development of the Bebras challenge. The questions of the contest are called "tasks". The tasks are created to be short, answerable in a few minutes through a computerized interface, and requiring deep-thinking skills in the informatics field [7]. Tasks are either multiple choice (four-choice questions with one correct answer) or interactive (using drag-and-drop techniques, assembling constructions, picking items, writing, etc.) [6]. The tasks are created in English. Each national organizer needs to translate the tasks from English into the national language spoken in his country.

The tasks are grouped into three categories which present increasing difficulties. Furthermore, each task of a category receives different points. Tasks of the "A" category receives +6 points for a correct answer and 0 points for an incorrect answer; tasks of the "B" category receives +9 points for a correct answer and -2 points for an incorrect answer; at last, tasks of the "C" category receives +12 points for a correct answer and -4 points for an incorrect answer (see Table I). This classification is made by the stakeholders.

IV. Method

This section describes instruments, participants and procedures adopted in our study.

A. Instruments

Until 2016 Brazil had not yet participated in the Bebras challenge. As Bebras occurs annually in several other countries and there are not tasks available in Portuguese, we chose tasks from other countries and over past years. We chose the English language to make it easier to translate the tasks into Portuguese, since the authors and other researchers who help us already knew English.

Among all the English-speaking countries that have participated in Bebras, the United Kingdom (UK) has produced annual brochures of the Bebras challenge tasks along with solutions and explanations. So, we selected the tasks of our study from the UK's Bebras test applied in 2014 and 2015. Moreover, we considered the senior level because it is the older age group.

Thus, the UK's Bebras test from 2014 and 2015 were translated into Portuguese. For reduce bias and translation mistakes, each task was reviewed by two external researchers from our study. From now on, test 1 means UK's Bebras applied in 2014 and test 2 means the test applied in 2015. In our study, the tasks of each test were randomly organized.

The student's absolute score in Bebras can vary from 0 to 15 (each test has fifteen tasks). The Bebras' score obeys the task categories presented in Table I. Although the students may answer all tasks incorrectly, we decide to work only with a positive score. So, the Bebras' score ranges from 0 to 135.

B. Participants

The participants of our study were CS undergraduates who attended the introductory programming course at two Brazilian universities during the second semester of 2016. Both universities use the Python programming language.

Two simulated multiple choice Bebras tests were applied. The first simulated multiple choice Bebras test was answered by 160 students while the second one was answered by 81 students. The difference in the number of students is due to many reasons: some students dropped out during the semester and others did not appear at the test day due to unforeseen circumstances. Because of these reasons, the number of students varies along each step of our investigation.

C. Procedures

The students participated in two simulated multiple choice Bebras tests. The first was applied at the beginning of the term and therefore before any exposition to programming classes. The second test was applied at the end of the term. In all cases, the authors supervised the performance of the test.

The students carried out the tests with paper and pencil in both times. In the beginning, they received basic instructions such as the duration time (55 minutes), tasks type, and the challenging context.

After all, the dataset was organized. The tests scores were storage in a table. Besides, the final grades on the introductory programming course were also included in the table. These grades are ranged from 0 to 10 in both universities. The data were analyzed by descriptive and inferential statistics using the R programming language. The results are reported in next section.

V. Results

This section presents the result of our statistical analysis with the collected data. First, we simply scored (absolute score) each correct task with 1 point, incorrect tasks with 0 points, and empty tasks were marked with N/A. When looking at the mean of absolute scores, the students reached 9.35 in test 1 (standard deviation of 2.05), and mean of 9.23 in test 2 (standard deviation of 3.07). In the introductory programming course, they reached mean of 7.45 (standard deviation of 2.06).

After that, we calculated the Bebras' score following the three difficulty categories presented in Table I. Figure 1 provides a summary of data. Apart from some outliers, the second boxplot presents a wider variation when compared to the first boxplot. Table II shows the means of absolute and Bebras' score. In both tests, the mean of absolute score is almost the same. Whereas, test 2 presents higher mean of Bebras' score and standard deviation than test 1.

TABLE II
Mean of Score and points

	Absolute Score	SD	Bebras' Score	SD
Test 1	8.9	2.4	100	24.1
Test 2	9.0	3.1	108	32.4

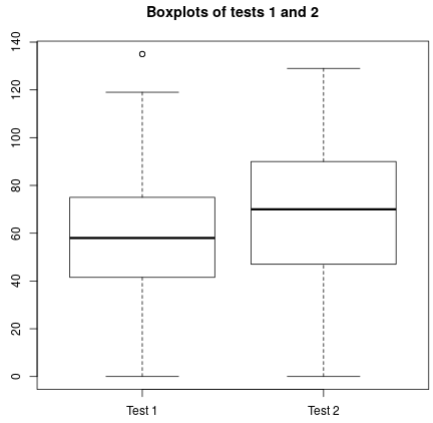

Fig. 1. Boxplots of Tests 1 and 2

Next, our data set were checked with respect to the distribution. The Shapiro-Wilk test was applied, with significance level of 0.05, in order to observe whether the data set follows a normal distribution. We obtained a *p-value* $= 0.319$ for test 1 and a *p-value* $= 0.264$ for test 2.

Thus, we can conclude that both samples follow a normal distribution with a confidence level of 95%. On the other hand, the result of the normality test for the introductory programming course grades obtained a *p-value* < 0.001, which means the grades are not normally distributed.

A. Is Bebras performance a good predictor of success for students within programming courses?

This section reports the results used to answer our research question 1: Is Bebras performance a good predictor of success for students within programming courses? To answer this question, we investigate the correlation between Bebras' score of test 1 and the introductory programming grades. In this case, 138 students did the test 1 and concluded the introductory programming course. In this step of our investigation, the sample is composed of $n = 138$ students.

Figure 2 shows a scatter plot that relates the Bebras' score of test 1 with the introductory programming grades. The red line indicates a positive correlation, i.e., as one variable increases, the other also increases, but the data seem to be very spread. So, considering that grades don't follow a normal distribution, we calculated the Spearman's rank correlation coefficient to

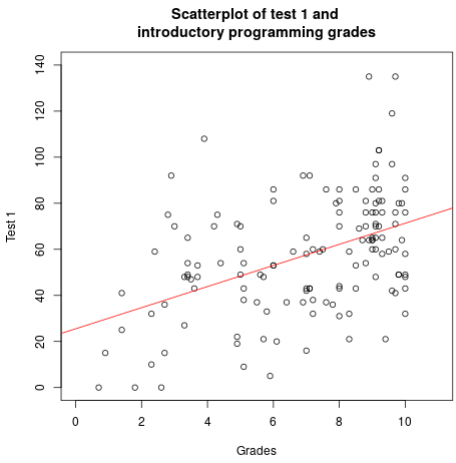

Fig. 2. Scatter plot of the Bebras' score of test 1 and the introductory programming grades

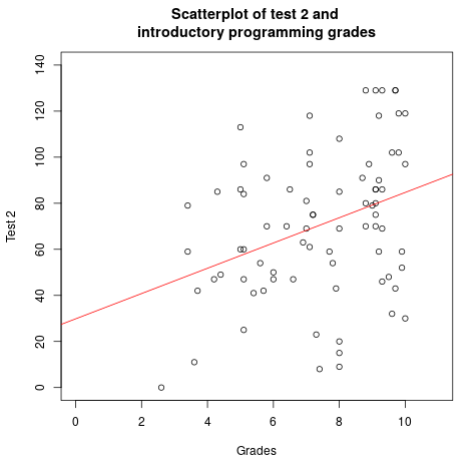

Fig. 3. Scatterplot of test 2 and the introductory programming grades

TABLE III
SPEARMAN'S RANK CORRELATION COEFFICIENT

Bebras' score	Grades
Test 1	0.425
Test 2	0.302

measure the strength of the correlation between the Bebras' score of test 1 and the introductory programming grades. The result was $\rho = 0.425$, as seen in Table III. This result confirms the moderate positive correlation, since $0.30 < \rho < 0.70$. Based on our data set, we can conclude that Bebras performance can moderately predict the students' success in the introductory programming course.

B. Is there any correlation between Bebras performance and students' grades?

This section reports the results obtained to answer our research question 2: Is there any correlation between Bebras performance and students' grades? Once we already find that test 1 has a moderate correlation with introductory programming grades, we now investigate whether test 2 is also correlated with introductory programming grades. In this time, 78 students did the test 2 and concluded the introductory programming course ($n = 78$).

Figure 3 shows the scatter plot that relates the Bebras' score of test 2 with the introductory programming grades. We also calculated the Spearman's rank correlation coefficient to measure the strength of the correlation. The result is $\rho = 0.302$, as shown in Table III. The correlation is moderate positive, since $0.30 < \rho < 0.70$. So, we can conclude that Bebras performance has moderate correlation with the introductory programming grades, independently if the test was administrated at the beginning or the end of the term.

C. Do students improve their performance in Bebras tests when exposed to the contents of a programming course?

The previous results show that Bebras has a moderate correlation with introductory programming. Now, we investigated if the students performance in Bebras improve along the programming course, i.e., whether the students improve their achievements in Bebras scores after the programming classes. In this case, our sample reduces to 69 students who carried out both tests 1 and 2 ($n = 69$).

To analyze the effect size of the mean difference, we calculated the Cohen's d. The result was $d = 2.344$ which means a large effect. Considering the effect size, we have evidence to believe the mean differs. However, to support this evidence, we need to perform a hypothesis test.

For choose the appropriate statistical test, we analyzed the data distribution again considering $n = 69$. The Shapiro-Wilk test was executed and the results indicate that test 1 (*p-value* $= 0.028$) does not have a normal distribution while test 2 is normally distributed (*p-value* $= 0.343$), with the confidence level of 95%. Therefore, we need a non-parametric test to assess whether the Bebras' scores differ from test 1 to test 2.

We conducted the Wilcoxon signed-rank test to analyze if the difference of mean of Bebras' score is statistically significant. We test the null hypothesis (the mean of test 1 and test 2 are the same), considering a significance level of 0.05. The result shows *p-value* $= 0.063$. Thus, we accept the null hypothesis that the mean of Bebras' score from test 1 and test 2 are the same.

VI. DISCUSSION

In this section, we discuss the previous results and highlight some issues that threaten the validity of our study.

A. Bebras performance is a moderate predictor of success for students within programming courses

We already expected that students without any pre-exposition of programming classes could achieve positive

results, since the tasks do not require previous knowledge of Computer Science to answer correctly. Besides, solving problems is a general skill not exclusively from computer scientists [1]. Furthermore, a recent study supports the statement that CT is fundamentally linked with general mental ability [21].

We also presumed student with higher Bebras' scores in test 1 also achieve higher grades in the introductory programming course and vice versa. However, the results of our dataset showed that Bebras are only moderately related to success within programming course. Thus, we cannot support that Bebras is a predictor of success in the introductory programming course. For the case when the students achieve higher Bebras' score and lower grades, the results of Bebras is not decisive to conclude that the students will make an excellent performance in the programming class. In the other hand, for the case when the students achieve higher grades and lower Bebras's score, we believe that learning programming requires others abilities beyond CT. In this controversial situations, we need further investigation to make some assumption.

B. There is a moderate correlation between Bebras performance and students' grades

The findings suggest that test 2, which was applied at the end of the term, has a moderate correlation with introductory programming grades. At first, we expected a stronger correlation between Bebras performance and grade because the studies have pointed out programming as the main approach to foster CT [2]. Examining the literature more carefully, however, we found studies that did not find any correlation between the performance of simulated multiple choice Bebras and grades by first-year software engineering undergraduate [10] or they found a weak correlation between Bebras performance and programming test for primary school students [5].

These controversial results incite questions and further investigation. In one direction, we regard some consideration over Bebras as an instrument to assess CT. New task sets are created annually, so, the Bebras tests applied in studies perhaps differ by difficulty level and specific CT abilities, such as abstraction, algorithm thinking, decomposition, evaluation, and generalization. Besides, the Bebras' score adopted in the contest could influence the overall result. In another direction, we should investigate the relation between CT and programming. Is it true that every programming course will improve CT abilities? Moreover, do the students who achieve higher scores in CT must achieve higher scores in programming course? Does CT define programming abilities and vice-versa?

C. Students do not improve their performance in Bebras tests

Considering that programming is the main approach to booster CT [2], we expected a significant difference in Bebras performance when comparing the beginning and the end of the term. In fact, the Cohen's d showed a large effect size ($d = 2.344$), supporting the statement that programming course stimulates CT abilities. By contrast, the hypothesis test had not presented evidence to reject the null hypothesis, i.e. the sample mean does not differ, at 95% confidence level (*p-value* $= 0.063$). However, if we consider 90% confidence level, we could reject the null hypothesis and assumed that there is a difference between sample means. So, there is a small threshold considering the confidence level to reject or not the null hypothesis.

This finding needs further investigation in two directions. First, we wonder if an introductory programming course is enough time to reveal differences in a test to measure latent variables. Second, we do not know if Bebras can be used as the psychometric instrument to measure CT abilities. Considering the last issue, we begin an exploratory study of the applicability of Item Response Theory methods to the Bebras. Next section discusses the evidence that Bebras can be used as a psychometric instrument to measure CT abilities.

D. Threats to validity

Some issues threaten the validity of our study. The study was limited to the number of participants at two universities, so we do not know whether the findings can be generalized to other CS courses. In addition, we analyzed the data of the two universities together. Even if we analyze the data separately by university, the overall result does not change, i.e., we can neither support that Bebras is a predictor of success in the introductory programming course nor students improve their performance in Bebras tests. A second limitation is that our data were collected at the beginning and at the end of the term, with no data collection taking place at another moment during the term. Future studies can benefit from collecting qualitative data during the term as well. Third, we did not consider retention factors in the introductory programming courses. At last, the scores were produced manually, for this reason, human factors threaten its validity. To mitigate this issue, we double-check the dataset.

VII. Exploring Item Response Theory

In this section, we discuss a brief analysis of Item Theory Response (ITR) under Bebras' tests. We aim at analyzing this theory with the intention of exploring a suitable way to measure CT. First, we present the ITR as well as the primary criteria of discrimination, difficulty, and answering correctly by guessing. Then, we describe the fundamental equation of the used psychometric model. Finally, we show the result of the two simulated Bebras tests.

According to psychometrics, for a test to measure latent variables, some criteria must be examined. The first is the difficulty of the item. The difficulty of an item works to position the item along the ability scale. For example, an easy item works among the low-ability examinees, and a hard item works among the high-ability examinees [22]. In Bebras, the tasks are classified according to their difficulty level. The researchers use their tacit knowledge to rank tasks in easy, medium or hard tasks. The issue to use this approach is that the difficulty classified by researchers can be overestimated or underestimated because they are not based on students' answers. The previous Bebras studies already have shown that

in one-third of the cases the tasks were either easier or harder than expected by researchers [23], and 13% to 60% of the tasks were easier or harder than predicted by researchers [8]. This approach can be sufficient for the Bebras original proposal as a contest to promote CT. However, when we examine it as an instrument of assessment, this approach does not appear to be the best choice.

Besides the difficulty, another two important criteria for psychometrics are discrimination of a question and answer correctly by guessing. Discrimination describes how well a question can differentiate students with higher and lower levels of knowledge [22]. In general, items with higher discrimination detects a small change in the ability level of students. At last, in tests with multiple choice questions, there is always the possibility of correctly guessing when giving a random answer. Thereby, the correct response includes a small probability of hit due to guessing. Assessing abilities by multiple choice questions must consider this case. To the best of our knowledge, Bebras does not examine the criteria of discrimination and hit due to guessing. Besides, the difficulty level is classified by researchers, not over the student's answers.

Thus, one way to use Bebras as an instrument of assessing CT, through psychometrics, is exploring IRT. IRT is a testing model which is based on individuals performances on a test designed to measure certain abilities [22]. To analyze multiple-choice items marked as correct or incorrect and which seem to allow guessing, the psychometrics uses the three-parameter logistic model (3PL). The fundamental equation of the 3PL model 1 is the probability that a randomly selected examinees with proficiency Θ will correctly respond to item j, characterized in terms of a is the slope of parameter of item, characterizing its sensitivity to proficiency; b is the threshold parameter of item j, characterizing its difficulty and c is the lower asymptote parameter of item j, reflecting the chances of students with very low proficiency selecting the correct response (guessing) [22].

$$P(\Theta) = c_j + (1 - c_j)\frac{1}{1 + e^{-a_j(\Theta - b_j)}} \quad (1)$$

Now we present a brief exploratory case of the applicability of IRT parameter to the simulated multiple choice Bebras test. Since IRT considers that only one latent ability is measured by a set of items in a test, we assumed that Bebras test measures Computational Thinking ability. Thus, the presence of CT ability as a dominant factor explains most of the expected instrument variance scores.

Considering IRT, Item Curve Characteristic (ICC) allows the graphic visualization of difficulty, discrimination and hit due to guessing parameters. ICC is an increasing monotonic mathematical function that predicted the behavior of the individual in an item. The S-shaped curve describes the relationship between the probability of a correct response (axis Y) given to an item and the ability scale (axis X) [22]. Each item in a test will have its ICC. Figures 4 and 5 provide the ICC for test 1 and test 2, respectively, considering Baye's modal estimator

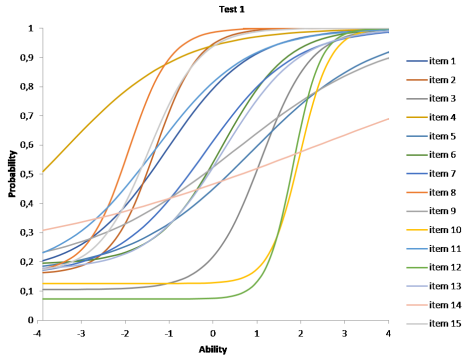

Fig. 4. Item Characteristic Curves of Test 1

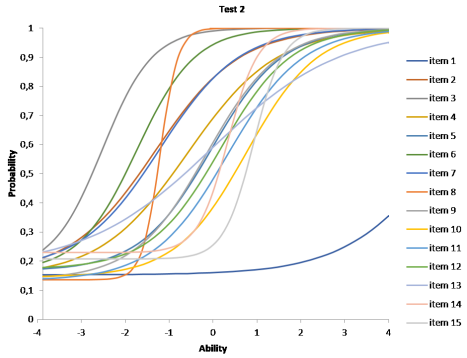

Fig. 5. Item Characteristic Curves of Test 2

and 3PL. For this purpose, we used the data set of all the students who answered each test: $n = 160$ for the test 1 and $n = 81$ for the test 2.

Observing Figures 4 and 5, we revisit difficulty, discrimination and hit due to guessing parameters for each test over ICC. The slope parameter reflects the steepness of the item and describes how many individuals of different abilities are distinguished as to the probability of hitting the item. So, the slope parameter specifies subjects with relative magnitudes in the latent trait, in our case, CT abilities. When discrimination is greater than moderate, the item characteristic curve is S-shaped and rather steep in its middle section. The items 12, 8, and 2 in Figure 4, and items 8, 3, and 14 in Figure 5 are examples of greater discrimination. Whereas, when the item discrimination is less than moderate, the item characteristic curve is nearly linear and appears rather flat. This occurs with items 14, 9, and 4 in Figure 4, and items 1, 4, and 13 in Figure 5. Notably, we found a critical value for the discrimination parameter in item 14 (Figure 4 - Test 1). The value of discrimination parameter of item 14 is lower than 0.30. This result indicates that the item should be review. Besides that, a particular case can be seen in item 14 in test 1 and item 1 in test 2. The flatter the curve, the less the item is able to discriminate since the probability of correct response at low ability levels is nearly the same as it is at high ability levels. In other words, if an item is either very

easy or very hard, it is not likely to be discriminating [22]. In this case, the item 14 in test 1 was answered correctly by a lot of students, while the item 1 in test 2 was answered correctly by few students. Because of that, the items are not helpful to discriminate students with higher and lower levels of CT ability.

For observing the difficulty parameter over ICC, we must see the value of $P(\Theta)$. The value of $P(\Theta) = 0.5$ corresponds to the item difficulty, i.e., b threshold parameter [22]. When an item is easy, this occurs at a low ability level. This occurs with items 4, 8 and 15 in test 1 and items 3, 6 and 2 in test 2. Meanwhile, when an item is hard, this corresponds to a high ability level. This occurs with items 10, 12 and 3 in test 1 and items 15, 10 and 14 in test 2. The probability of hit to guessing is below 0.40 for both tests, so we do not found any critical values for this parameter.

Finally, we investigate the reliability of the both tests. We calculate the Cronbach's Alpha, a measure for estimating the reliability of a test. The Cronbach's Alpha evaluates a magnitude in which the items of an instrument are correlated, i.e., the mean of the correlations between the items that are part of an instrument [24]. Cronbach's alpha is 0.56 for test 1 and 0.68 for test 2. These measures indicate "poor" and "questionable" internal consistency, respectively. These results suggest that the tasks should be better selected to improve internal consistency.

This Section presented a brief overview of IRT. The findings suggest that it is not very likely that CT measures can be derived from the Bebras test as it is currently designed. Nevertheless, we believe that it is possible to take advantage of the international effort to extend Bebras proposal as an assessment instrument. Further investigation is necessary for exploring others aspects of IRT.

VIII. Concluding remarks

In this exploratory and preliminary study, we investigate how appropriate a Bebras challenge is as an instrument to assess and measure CT abilities. First, we investigated whether Bebras performance a good predictor of success for students within programming courses. Our dataset suggests that Bebras performance can moderately predict the students' success of introductory programming course. Second, we investigated the correlation between Bebras performance at the end of the term and students grades. We also find a moderate correlation when multiple choice Bebras test was applied at the end of the term. Third, we investigated whether students improve their performance in Bebras tests when exposed to the contents of a programming course. The result shows a large effect size (d = 2.344) but it does not show statistically significant difference (Wilcoxon signed-rank test *p-value* = 0.063).

As highlighted by Dagienė and Stupurienė [4], the main goal of the Bebras challenge is not to test students knowledge, but with more investigation, it looks like that Bebras can be used for assessment. In this direction, we presented a brief study of the applicability of Item Response Theory parameters to Bebras. The finding suggests that it is not very likely that CT measures can be derived from the Bebras test as it is currently designed. Further research is needed on how we can leverage the Bebras effort to extend it into a CT assessment instrument. Nevertheless, we believe that it is possible to take advantage of this international effort (Bebras) in order to produce an adequate assessment instrument of CT abilities.

As future work, we plan to deepen our research of IRT to Bebras. We will investigate the qualitative aspects involved in each question that can indicate the discrimination and the difficulty level of each question used in this study. We expect the results of this research can contribute to both the development and discussion of CT assessment as well as the Bebras effort to disseminate CT.

References

[1] J. M. Wing, "Computational thinking," *Communications of the ACM*, vol. 49, no. 3, pp. 33–35, 2006.

[2] A. L. S. O. de Araujo, W. L. Andrade, and D. D. S. Guerrero, "A systematic mapping study on assessing computational thinking abilities," in *Frontiers in Education Conference (FIE), 2016 IEEE*. IEEE, 2016, pp. 1–9.

[3] J. P. Guilford, *Psychometric methods*. McGraw-Hill, 1954.

[4] V. Dagienė and G. Stupuriene, "Bebras-a sustainable community building model for the concept based learning of informatics and computational thinking," *Informatics in Education-An International Journal*, no. Vol15_1, pp. 25–44, 2016.

[5] C. Duncan and T. Bell, "A pilot computer science and programming course for primary school students," in *Proceedings of the Workshop in Primary and Secondary Computing Education*. ACM, 2015, pp. 39–48.

[6] V. Dagiene, L. Mannila, T. Poranen, L. Rolandsson, and P. Söderhjelm, "Students' performance on programming-related tasks in an informatics contest in finland, sweden and lithuania," in *Proceedings of the 2014 conference on Innovation & technology in computer science education*. ACM, 2014, pp. 153–158.

[7] V. Dagienė, E. Pelikis, and G. Stupurienė, "Introducing computational thinking through a contest on informatics: Problem-solving and gender issues." *Informacijos Mokslai/Information Sciences*, vol. 73, 2015.

[8] W. Van der Vegt, "Predicting the difficulty level of a bebras task," *Olympiads in Informatics*, vol. 7, pp. 132–139, 2013.

[9] C. Izu, C. Mirolo, A. Settle, L. Mannila, and G. Stupurienė, "Exploring bebras tasks content and performance: A multinational study," *Informatica*, vol. 16, no. 1, pp. 39–59, 2017.

[10] V. Dolgopolovas, T. Jevsikova, L. Savulionienė, and V. Dagienė, "On evaluation of computational thinking of software engineering novice students," in *Proceedings of the IFIP TC3 Working Conference A New Culture of Learning: Computing and next Generations*, 2015, pp. 90–99.

[11] A. Basawapatna, K. H. Koh, A. Repenning, D. C. Webb, and K. S. Marshall, "Recognizing computational thinking patterns," in *Proceedings of the 42Nd ACM Technical Symposium on Computer Science Education*, ser. SIGCSE '11. New York, NY, USA: ACM, 2011, pp. 245–250.

[12] K. H. Koh, H. Nickerson, A. Basawapatna, and A. Repenning, "Early validation of computational thinking pattern analysis," in *Proceedings of the 2014 Conference on Innovation & Technology in Computer Science Education*, ser. ITiCSE '14. New York, NY, USA: ACM, 2014, pp. 213–218.

[13] K. Brennan and M. Resnick, "New frameworks for studying and assessing the development of computational thinking," in *Proceedings of the 2012 annual meeting of the American Educational Research Association, Vancouver, Canada*, 2012.

[14] L. Seiter and B. Foreman, "Modeling the learning progressions of computational thinking of primary grade students," in *Proceedings of the ninth annual international ACM conference on International computing education research*. ACM, 2013, pp. 59–66.

[15] I. Zur-Bargury, B. Pârv, and D. Lanzberg, "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*, ser. ITiCSE '13. New York, NY, USA: ACM, 2013, pp. 267–272.

[16] S. Grover, "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. AERA'15, Chicago, USA*, 2015.

[17] S. Grover, S. Cooper, and R. Pea, "Assessing computational learning in k-12," in *Proceedings of the 2014 Conference on Innovation ; Technology in Computer Science Education*, ser. ITiCSE '14. New York, NY, USA: ACM, 2014, pp. 57–62.

[18] L. Gouws, K. Bradshaw, and P. Wentworth, "First year student performance in a test for computational thinking," in *Proceedings of the South African Institute for Computer Scientists and Information Technologists Conference*, ser. SAICSIT '13. New York, NY, USA: ACM, 2013, pp. 271–277.

[19] C. Marais and K. Bradshaw, "Problem-solving ability of first year cs students: A case study and intervention," in *Proceedings of the 44th Conference of the Southern African Computers Lecturers' Association*, 2015.

[20] V. Dagienė, S. Sentance, and G. Stupurienė, "Developing a two-dimensional categorization system for educational tasks in informatics," *Informatica*, vol. 28, no. 1, pp. 23–44, 2017.

[21] M. Román-González, J.-C. Pérez-González, and C. Jiménez-Fernández, "Which cognitive abilities underlie computational thinking? criterion validity of the computational thinking test," *Computers in Human Behavior*, 2016.

[22] F. B. Baker, *The basics of item response theory*. ERIC, 2001.

[23] C. Bellettini, V. Lonati, D. Malchiodi, M. Monga, A. Morpurgo, and M. Torelli, "How challenging are bebras tasks?: an irt analysis based on the performance of italian students," in *Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education*. ACM, 2015, pp. 27–32.

[24] D. L. Streiner, "Being inconsistent about consistency: When coefficient alpha does and doesn't matter," *Journal of personality assessment*, vol. 80, no. 3, pp. 217–222, 2003.