Computers in Human Behavior xxx (2016) 1–14

Contents lists available at ScienceDirect

Computers in Human Behavior

journal homepage: www.elsevier.com/locate/comphumbeh

Full length article

Which cognitive abilities underlie computational thinking? Criterion validity of the Computational Thinking Test

Marcos Román-González*, Juan-Carlos Pérez-González, Carmen Jiménez-Fernández

Universidad Nacional de Educación a Distancia (UNED), Faculty of Education, C/ Juan del Rosal, n° 14, C.P. 28040, Madrid, Spain

ARTICLE INFO

Article history:
Received 17 April 2016
Received in revised form
22 July 2016
Accepted 29 August 2016
Available online xxx

Keywords:
Computational thinking
Computational Thinking Test
Code literacy
Computer science education
Cognitive abilities
Cognitive assessment
Educational psychology
Primary education
Secondary education

ABSTRACT

Computational thinking (CT) is being located at the focus of educational innovation, as a set of problem-solving skills that must be acquired by the new generations of students to thrive in a digital world full of objects driven by software. However, there is still no consensus on a CT definition or how to measure it. In response, we attempt to address both issues from a psychometric approach. On the one hand, a Computational Thinking Test (CTt) is administered on a sample of 1,251 Spanish students from 5th to 10th grade, so its descriptive statistics and reliability are reported in this paper. On the second hand, the criterion validity of the CTt is studied with respect to other standardized psychological tests: the Primary Mental Abilities (PMA) battery, and the RP30 problem-solving test. Thus, it is intended to provide a new instrument for CT measurement and additionally give evidence of the nature of CT through its associations with key related psychological constructs. Results show statistically significant correlations at least moderately intense between CT and: spatial ability ($r = 0.44$), reasoning ability ($r = 0.44$), and problem-solving ability ($r = 0.67$). These results are consistent with recent theoretical proposals linking CT to some components of the Cattel-Horn-Carroll (CHC) model of intelligence, and corroborate the conceptualization of CT as a problem-solving ability.

1. Introduction

We live immersed in a digital ecosystem full of objects driven by software (Manovich, 2013). In this context, being able to handle the language of computers is emerging as an inescapable skill, a new literacy, which allows us to participate fully and effectively in the digital reality that surrounds us: it is about to *'program or be programmed'* (Rushkoff, 2010); it is about to be *'app-enabled or app-dependent'* (Gardner & Davis, 2013). The term 'code-literacy' has recently been coined to refer to the process of teaching and learning to read-write with computer programming languages (Prensky, 2008; Rushkoff, 2012). Thus, it is considered that a person is code-literate when is able to read and write in the language of computers and other machines, and to think computationally (Román-González, 2014). If code-literacy refers ultimately to a new read-write practice, computational thinking (CT) refers to the underlying problem-solving cognitive process that allows it. In other words, computer programming is the fundamental way that enables CT come alive (Lye & Koh, 2014); although CT can be transferred to various types of problems that do not directly involve programming tasks (Wing, 2008).

Given this current reality overrun by the digital, it is not surprising that there is renewed interest in many countries to introduce CT as a set of problem-solving skills to be acquired by the new generations of students; even more, CT is becoming viewed at the core of all STEM (*Science, Technology, Engineering, & Mathematics*) disciplines (Henderson, Cortina, & Wing, 2007; Weintrop et al., 2016). Although learn to think computationally has long been recognized as important and positive for the cognitive development of students (Liao & Bright, 1991; Mayer, 1988; Papert, 1980), as computation has become pervasive, underpinning communication, science, culture and business in our society (Howland & Good, 2015), CT is increasingly seen as an essential skill to create rather than just consume technology (Resnick et al., 2009). Thus, many governments around the world are incorporating computer programming into their national educational curricula. The recent decision to introduce computer science teaching from primary school onwards in the UK (Brown et al., 2013) and others European countries (European Schoolnet, 2015) reflects the growing recognition of the importance of CT.

* Corresponding author.
E-mail addresses: mroman@edu.uned.es (M. Román-González), jcperez@edu.uned.es (J.-C. Pérez-González), mjimenez@edu.uned.es (C. Jiménez-Fernández).

http://dx.doi.org/10.1016/j.chb.2016.08.047
0747-5632/© 2016 Elsevier Ltd. All rights reserved.

However, there is still little consensus on a formal definition of CT (Gouws, Bradshaw, & Wentworth, 2013; Kalelioğlu, Gülbahar, & Kukul, 2016), and disagreements over how it should be integrated in educational curricula (Lye & Koh, 2014). Similarly, there is a worrying vacuum about how to measure and assess CT, fact that must be addressed. Without attention to assessment, CT can have little hope of making its way successfully into any curriculum. Furthermore, in order to judge the effectiveness of any curriculum incorporating CT, measures that would enable educators to assess what the student has learned need to be validated (Grover & Pea, 2013).

In response, we attempt to address these issues from a psychometric approach. On the one hand, how our Computational Thinking Test (CTt) has been designed and developed is reported, as well as its descriptive statistics and reliability derived from an administration on a sample exceeding a thousand Spanish students. On the other hand, the criterion validity (Cronbach & Meehl, 1955) of the CTt is studied with respect to already standardized psychological tests of core cognitive abilities. Thus, this paper is aimed at providing a new instrument for measuring CT and additionally giving evidence of the correlations between CT and other well-established psychological constructs in the study of cognitive abilities.

1.1. Computational thinking definitions

We can distinguish between: a) generic definitions; b) operational definitions; c) educational and curricular definitions.

1.1.1. Generic definitions

One decade ago, in 2006, Jeanette Wing's foundational paper defined that CT "involves solving problems, designing systems, and understanding human behavior, by drawing on the concepts fundamental to computer science" (Wing, 2006, p. 33). Thus, CT's essence is thinking like a computer scientist when confronted with a problem. But this first generic definition has been revisited and specified in successive attempts over the last few years, still not reaching an agreement (Grover & Pea, 2013; Kalelioğlu et al., 2016). So, in 2011 Wing clarified, CT "is the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent" (Wing, 2011; on-line). One year later, this definition is simplified by Aho, who conceptualizes CT as the thought processes involved in formulating problems so "their solutions can be represented as computational steps and algorithms" (Aho, 2012, p. 832).

1.1.2. Operational definitions

In 2011, the Computer Science Teachers Association (CSTA) and the International Society for Technology in Education (ISTE) developed an operational definition of computational thinking that provides a framework and common vocabulary for Computer Science K-12 educators: CT is a "problem-solving process that includes (but is not limited to) the following characteristics: formulating problems in a way that enables us to use a computer and other tools to help solve them; logically organizing and analyzing data; representing data through abstractions such as models and simulations; automating solutions through algorithmic thinking (a series of ordered steps); identifying, analyzing, and implementing possible solutions with the goal of achieving the most efficient and effective combination of steps and resources; generalizing and transferring this problem solving process to a wide variety of problems" (CSTA & ISTE, 2011; on-line).

1.1.3. Educational-curricular definitions

More than definitions in the strict sense, frameworks for developing CT in the classroom and other educational settings are mentioned next. So, from the UK, the organization Computing At School (CAS) states that CT involves six different concepts (logic, algorithms, decomposition, patterns, abstraction, and evaluation), and five approaches to working (tinkering, creating, debugging, persevering, and collaborating) in the classroom (CAS Barefoot, 2014). Moreover, from the United States, Brennan and Resnick (2012) describe a CT framework that involves three key dimensions: 'computational concepts' (sequences, loops, events, parallelism, conditionals, operators, and data); 'computational practices' (experimenting and iterating, testing and debugging, reusing and remixing, abstracting and modularizing); and 'computational perspectives' (expressing, connecting, and questioning). Table 1 shows a crosstab intersecting the CT framework dimensions (Brennan & Resnick, 2012) with the sampling domain of our Computational Thinking Test (CTt), which will be detailed in Sub-section 1.4.

1.2. Computational thinking from the CHC model of intelligence

While CT involves thinking skills to solve problems algorithmically (e.g., Brennan & Resnick, 2012; Grover & Pea, 2013), intelligence (i.e., general mental ability or general cognitive ability) involves primarily the ability to reason, plan and solve problems (Gottfredson, 1997). Even authors with alternative approaches to the conceptualization of intelligence recognize intelligence as a "computational capacity" or "the ability to process certain kinds of information in the process of solving problems of fashioning products" (Gardner, 2006, p. 503).

Within a cognitive approach, it has been recently suggested (Ambrosio, Xavier, & Georges, 2014) that computational thinking is related to the following three abilities-factors from the Cattell-Horn-Carroll (CHC) model of intelligence (McGrew, 2009; Schneider & McGrew, 2012):

- **Fluid reasoning (G_f)**, defined as: "the use of deliberate and controlled mental operations to solve novel problems that cannot be performed automatically. Mental operations often include drawing inferences, concept formation, classification, generating and testing hypothesis, identifying relations, comprehending implications, problem solving, extrapolating, and transforming information. Inductive and deductive reasoning are generally considered the hallmark indicators of G_f" (McGrew, 2009, p. 5)
- **Visual processing (G_v)**, defined as "the ability to generate, store, retrieve, and transform visual images and sensations. G_v abilities are typically measured by tasks (figural or geometric stimuli) that require the perception and transformation of visual shapes, forms, or images and/or tasks that require maintaining spatial orientation with regard to objects that may change or move through space" (McGrew, 2009, p. 5)
- **Short-term memory (G_{sm})**, defined as "the ability to apprehend and maintain awareness of a limited number of elements of information in the immediate situation (events that occurred in the last minute or so). A limited-capacity system that loses information quickly through the decay of memory traces, unless an individual activates other cognitive resources to maintain the information in immediate awareness" (McGrew, 2009, p. 5).

Therefore, it is expected that a computational thinking test should correlate with other already validated tests aimed at measuring cognitive abilities cited above.

Please cite this article in press as: Román-González, M., et al., Which cognitive abilities underlie computational thinking? Criterion validity of the Computational Thinking Test, Computers in Human Behavior (2016), http://dx.doi.org/10.1016/j.chb.2016.08.047