

Assessing Product Creativity in Computing Education: A Systematic Mapping Study

Nathalia DA CRUZ ALVES, Christiane GRESSE VON WANGENHEIM,
Lúcia Helena MARTINS-PACHECO

*Department of Informatics and Statistics (INE) – Federal University of Santa Catarina (UFSC)
Florianópolis/SC, Brazil
e-mail: nathalia.alves@posgrad.ufsc.br, {c.wangenheim, lucia.pacheco}@ufsc.br*

Received: April 2020

Abstract. Creativity has emerged as an important 21st-century competency. Although it is traditionally associated with arts and literature, it can also be developed as part of computing education. Therefore, this article -presents a systematic mapping of approaches for assessing creativity based on the analysis of computer programs created by the students. As result, only ten approaches reported in eleven articles have been encountered. These reveal the absence of a commonly accepted definition of product creativity customized to computer education, confirming only originality as one of the well-established characteristics. Several approaches seem to lack clearly defined criteria for effective, efficient and useful creativity assessment. Diverse techniques are used including rubrics, mathematical models and machine learning, supporting manual and automated approaches. Few performed a comprehensive evaluation of the proposed approach regarding their reliability and validity. These results can help instructors to choose and adopt assessment approaches and guide researchers by pointing out shortcomings.

Keywords: creativity, product creativity, assessment, computing, programming, systematic mapping.

1. Introduction

In today's rapidly changing digital society, creativity is considered one of the main 21st century competencies essential for professional and personal success (Kaufman and Beghetto, 2009; Voogt and Roblin, 2012). Consequently, developing students' creativity from an early age has become a dominant concern (Voogt and Roblin, 2012; Beghetto, 2010). Diverse curriculum frameworks also explicitly express the need for K-12 schools to foster creativity (P21, 2020; ISTE, 2020; Voogt and Roblin, 2012). Supplying students with opportunities to engage in creative ways can help them to develop the capacities to undertake work that cannot easily be automated and address increasingly complex challenges with out-of-the-box solutions (Sternberg, 2015).

And, although, creativity is traditionally associated with arts, music, and literature, it can also be developed as part of other knowledge areas, such as computing, for which design, research, and innovation are required (Bennett *et al.*, 2011). Teaching the solution of computational problems by creating novel and appropriate/useful computer programs can allow students to express their ideas (Romero *et al.*, 2017). Computing may also nurture competencies, such as imagination, visualization, abstraction, to solve problems creatively (Clements and Gullo, 1984; Yadav and Cooper, 2017; Grover and Pea, 2013) while, on the other hand, creative skills enhance solving algorithmic problems, creating computational artifacts, and developing new knowledge (Shell *et al.*, 2014). In K-12 education, this is typically introduced by teaching the development of computer programs using visual block-based programming languages, such as Scratch, BYOB/Snap! or App Inventor (Lye and Koh, 2014). In order to stimulate creativity, programming activities are often posed as open-ended ill-defined problems in a constructivist context adopting a problem-based and design-based learning strategy as “learning-by-making” (Bjögvinsson *et al.*, 2012).

Yet, a critical dimension of education is assessment in order to measure to which regard learning objectives have been achieved expressed as grades as well as a feedback mechanism to guide the learning and the teaching process. However, assessing creativity is challenging (Henriksen *et al.*, 2015). Especially in the context of problem-based learning, performance assessments can be conducted concerning the creative product as one crucial strand of creativity (Rhodes, 1961), representing the outcome of the learning process by the students (Ritchie, 2001). Such assessments typically measure whether the properties assigned to creative products are present in the outcomes and to what degree.

Although product creativity is commonly defined in terms of novelty and appropriateness (Jackson and Messick, 1964), thus, regarding how much a product differs from the norm and meets the practical needs of the problem situation, there exists no single definition of creativity. Frequently other terms such as style, including “organic, well-crafted, elegant” (O’Quin, 1987) and transformation, which refers to “some objects combine elements in ways that defy tradition” (Jackson and Messick, 1964) are also used in the definition of product creativity.

Yet, a question is which approaches exist for assessing creativity based on computer programs as outcomes of computing education. And, although there exist already a variety of reviews on creativity assessments, they are mostly targeted on general approaches to measure creativity and not focusing on specific strands (Said-Metwaly *et al.*, 2017). Bolden *et al.* (2019) and Snyder *et al.* (2019) provide overviews of the assessment of creativity in any discipline in K-12, not targeting specifically on computing education. Very few reviews are related to creativity in the context of computer science, analyzing approaches for leveraging creativity in agile requirements software engineering (Aldave *et al.*, 2019) and individual creativity support systems (Wang and Nickerson, 2017). However, these reviews in the computing context typically do not focus on an educational perspective nor specifically on the product.

On the other hand, reviews on assessments based on the analysis of computer programs typically focus only on computational thinking concepts, not covering creativity

(Araujo *et al.*, 2016; Alves *et al.*, 2019; Cutumisu *et al.*, 2019). Other studies such as Clements (1995), Scherer *et al.* (2019) focus on the effects on process creativity by learning computer programming, rather than concentrating on product creativity measurement. Typically, psychological tests such as the Torrance Test of Creative Thinking (TTCT) are used in these studies to assess creativity with respect to the creative process and not the product (Scherer *et al.*, 2019).

Thus, despite the recognition of the importance of assessing creativity in the context of the creative product created as an outcome of computing education, a detailed overview on how to design such assessments in a reliable, valid way useful to both learners and teachers is currently inexistent. Therefore, we aim at advancing the understanding of how assessment can support and promote creativity in the classroom before mentioned context by performing a systematic mapping study. The results of the study can help instructors to systematically choose and adopt effective assessment approaches, as well as guide researchers by pointing out shortcomings in the existing approaches.

2. Background

2.1. Measuring Creativity

While there are many definitions of creativity, there are also disagreements leading to a lack of a standard definition and an inconsistent understanding (Walia, 2019). Yet, as creativity is multidimensional and can be represented from different perspectives, the way it is defined influences how to measure and assess creativity (Walia, 2019).

Creativity is generally defined in terms of the capacity to generate new, original, or surprising ideas and solutions (Walia, 2019). A creative idea or product is considered original if it represents something novel or surprising that did not exist before (Runco and Jaeger, 2012). However, novelty alone does not make something creative. Creative ideas or products also have to include an underlying value and usefulness, providing solutions that are appropriate, functional, correct, and valuable (Jackson and Messick, 1964). Most definitions of creativity, thus, identify originality and appropriateness as key characteristics of creative outcomes (Besemer and Treffinger, 1981; Runco and Jaeger, 2012). Beyond these common characteristics, several definitions also consider additional elements. Examples include “wholeness”, considering aesthetic dimensions, situated within the work’s specific context as proposed by Mishra and Henriksen (2013), detail and elegance (Besemer and Treffinger, 1981), among others.

Aiming at a better understanding, diverse researches have also proposed ways to structure the definition of creativity. Among those is Rhode’s (1961) widely recognized Four P’s framework for creativity (Fig. 1):

- Person refers to the individual that is performing the creative act. This includes the personality and various traits and attitudes of the creative individual as well as the individual’s creative potential.

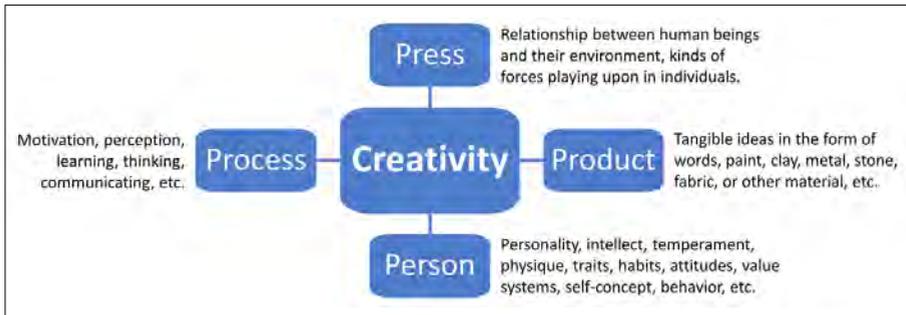


Fig. 1. 4Ps: Creativity strands (Rhodes, 1961).

- Process refers to the mental processes during creative endeavors involving the learning, thinking, and communication of ideas as well as the tools and strategies employed.
- Product refers to the tangible outcomes of the creative process (i.e., works of art, writings, computer programs, etc.).
- Press refers to the environment, setting, in which creativity takes place and whether the environment favors the relationship with the individuals regarding creativity.

While all four aspects of the framework play a role in understanding creativity, the analysis of the creative product as one strand may allow insights on the concept of creativity as a whole. Thus, although, the assessment of creativity cannot be confined to the point of view of the product (Ritchie, 2001), it certainly represents a vital part to provide “a means for establishing referents for the concept ‘creativity’ through a systematic evaluation of things which people produce” (Skager *et al.*, 1966).

2.1.1. Product Creativity

In this respect, a product-based approach measures whether the properties typically assigned to creative products are present in the outcomes and to what degree. Yet, again, there does not exist a conclusive set of these criteria, although the criteria for measuring creative products have been widely discussed (Besemer and Treffinger 1981; Ranjan *et al.*, 2018). Originality, appropriateness, and condensation are considered by most authors as the most important characteristics of a creative product, often using varying terms (Table 1).

There are others characteristics defined by some authors that represent extreme forms of the characteristics presented in Table 1, such as transformation, which refers to combining “elements in ways that defy tradition and that yield a new perspective” and “forces us to see reality in a new way” (Jackson and Messick, 1964). However, these can be grouped with the core characteristics, for example by understanding transformation as an extreme form of originality as being a revolutionary product “that it forces a shift in the way that reality is perceived” (Besemer and Treffinger, 1981).

Table 1
 Characteristics and other terms widely used for the definition of product creativity

Characteristic	Description	Synonyms used
Originality	refers to a product “level of surprisingness, and its projected germinal qualities (characteristics related to perceived influence in suggesting spin-offs or other new products)” (O’Quin, 1987). And to the extent of “newness of the product: in terms of the number and extent of new processes, new techniques, new materials, new concepts included” (Besemer and Treffinger, 1981).	Novelty; Unusualness; Surprisingness; Newness; Uniqueness; Germinal; Divergent; Infrequency; Revolutionary; Transformational
Appropriateness	“measures the extent to which a product meets the practical needs of the problem situation” (O’Quin, 1987). “To be appropriate a product must fit its context. It must “make sense” in light of the demands of the situation and the desires of the producer” (Jackson and Messick, 1964).	Resolution; Usefulness; Valuable; Adequate; Logical; Relevant; Significant; Meaningfulness; Workable; Worthiness; Effective; Correctness
Condensation	refers to “the fact that sometimes the initial design is elaborated and made more complex through working out the solution and other times (or simultaneously) the design may be refined and made simpler” (Besemer, 2000). “ It considers the aspects of style or production values” (O’Quin, 1987). “In the highest forms of creative condensation the polar concepts of simplicity and complexity are unified” (Jackson and Messick, 1964).	Elaboration and Synthesis; Style; Elegance; Completeness; Standardization; Sophistication; Subtle; Deft; Attractive; Engaging; Organic; Well-crafted; Complex; Care; Effort; Wholeness; Just right; Refined; Expressive

2.2. Assessment of Product Creativity

In everyday life, assessing creativity happens naturally, but in the classroom, it must move beyond such subjective measurements (Mishra and Henriksen, 2013). And, although some assessment approaches are considered “golden standards”, such as the Torrance Tests of Creative Thinking, a psychological measurement of an individual’s divergent thinking, they may provide little practical use in the classroom (Kaufman *et al.*, 2016). Especially when considering assessment in active learning environments using problem- and design-based strategies following a constructionist theory, it becomes clear that measuring the outcomes of these practical learning experiences plays a crucial role and has the potential to be highly authentic (Bialik *et al.*, 2016). In this context, an alternative widely used is the assessment of product creativity seeking to evaluate the outcomes of the creative process created by the students (Long, 2014; Mishra and Henriksen, 2017). These “products” in the context of computing education are typically computer programs, such as games or mobile applications.

This kind of assessment typically focuses on an analysis of the product to measure the creativity of the output by some standard, verifiable (reliable) measure (Table 2). Many approaches use a scale, such as the Student Product Assessment Form (SPAF) (Renzulli and Reis, 1991). The SPAF uses a 5-point ordinal rating scale (from poor to outstanding) to analyze creativity criteria related to quality, care, attention to detail, appropriateness, and originality (Renzulli and Reis, 1991). O’Quin and Besemer (1989)

Table 2
Characteristics related to product creativity in literature

Originality related characteristics	Appropriateness related characteristics	Condensation related characteristics	References
Novelty	Resolution	Elaboration and Synthesis	O'Quin and Besemer, 1989
Novelty	Resolution	Style	Besemer, 2000
Originality & Reformulation	Meaningfulness	Condensation	Haberland and Dacin, 1992
Unusualness & Transformation	Appropriateness	Condensation	Jackson and Messick, 1964
Infrequency		Complexity, Effort, Attractiveness	Ward and Warren, 1971
Originality	Technical goodness	Aesthetic appeal	Westberg, 1996
Originality of the Idea, Original Contribution	Achieved Objectives Stated in Plan, Advanced Familiarity with Subject	Care, Attention to Detail, Quality Beyond Age/Grade Level, Time, Effort, Energy	Renzulli and Reis, 1991
Novel	Effective	Whole	Mishra and Henriksen, 2013
Custom (expert panel)	Custom (expert panel)	Custom (expert panel)	Amabile, 1982

developed the Creative Product Semantic Scale based on the Creative Product Analysis Matrix (CPAM), considering three main criteria: novelty, resolution, and elaboration & synthesis. Mishra and Henriksen proposed a similar approach to assess novel, effective and whole (NEW) characteristics of creative products (Mishra and Henriksen, 2013).

Yet, in the context of open-ended assignments in problem- and design-based learning, these judgments become more complex and difficult, leaving educators to subjective assessments (Dousay, 2018). This may lead to inaccurate results, especially by interdisciplinary educators that may lack competence for accurate assessment of computer programs. In this context, one of the most popular is the Consensual Assessment Technique (CAT) (Amabile, 1982) that relies on a panel of expert judges (instructors or peers) to assess the creativity of the product and analyze several different characteristics depending on the panel aiming at consensus (Table 2).

Comparing existing product assessment approaches, differences concerning the assessment criteria become obvious (Long, 2014) (Table 2). However, the reliable and valid definition of these criteria is essential to measure creative products and identify what makes them different from non-creative products.

2.3. Creativity in Computing Education

While creativity is traditionally associated with arts, music, and literature, it can likewise be developed as part of other knowledge areas, such as computing (Bennett *et al.*, 2011).

Computing is recognized as a creative human activity that allows the exploration and creation of knowledge, enables innovation and allows individuals to deploy technology towards creating novel artifacts (Mishra and Yadav, 2013).

Teaching computing in school focuses typically on computational thinking aiming at expressing solutions as computational steps or algorithms that can be carried out by a computer (CSTA, 2016). It involves solving problems, designing systems, and understanding human behavior by drawing on the concepts fundamental to computer science (Wing, 2006). Computational thinking has also been linked to creativity and innovation, as it provides learners with the opportunity to express their ideas and create innovative programs to new and unexpected problems (DeSchryver and Yadav, 2015). Computational thinking may nurture competencies, such as imagination, visualization, abstraction, to creatively solve problems and applying computational thinking principles in problem-solving (Yadav and Cooper, 2017), while, on the other hand, creative skills enhance solving algorithmic problems, creating computational artifacts, and developing new knowledge (Shell *et al.*, 2014).

Moreover, the creative use of digital technologies to solve diverse problems engages students in an active design and creation process using computational concepts and methods (Romero *et al.*, 2017). By moving from computational thinking to computational making (Rode *et al.*, 2015), computing education in K-12 allows students to learn to create, test and refine computer programs (Shute *et al.*, 2017; CSTA, 2016; Lye and Koh, 2014), enabling them to creatively express themselves, concretize their ideas, and develop diverse and innovative ways to build and to learn (Clements, 1995; Grover and Pea 2013). In this respect, creativity is one of the keys to respond common challenges in the development of computer programs today (Robertson, 2005), as programming is not only about writing computer programs but also about competencies:

- To analyze context and requirements (Robertson, 2005).
- To ideate novel, useful and technically feasible solutions (Romero *et al.*, 2017).
- To design a computer program by modeling data and architecture (Gu and Tong, 2004).
- To design a usable and visually attractive user interface (Rode *et al.*, 2015; Ferreira *et al.*, 2019).
- To implement and test code (Glass, 1995).

Thus, in this context, creativity can be seen as an ability to apply imagination to create a computer program that is judged to be novel and appropriate, useful, and valuable providing solutions to practical problems (DeSchryver and Yadav, 2015).

Many computing education initiatives in K-12 introduce algorithms & programming concepts by using visual block-based programming languages, such as Scratch, BYOB/Snap! or App Inventor (Lye and Koh, 2014). Programming activities are often posed as open-ended ill-defined problems in a constructivist context adopting a problem-based and design-based learning strategy (Bjögvinsson *et al.*, 2012). Thinking of it as “learning-by-making” driven by the maker movement, this creative computing approach inspires learners to create their outcomes engaging them in the construction of digital and tangible artifacts through the use of technologies (Rode *et al.*, 2015;

Brennan *et al.*, 2019). And, in order to provide students with the opportunity to do computing in ways that have a direct impact on their lives and their communities, often a perspective of computational action (Tissenbaum *et al.*, 2019) is adopted focusing on real-world problems usually in an interdisciplinary way (Dousay, 2018). These activities aim to stimulate the development of higher-order thinking competencies not prescribing a correct or best solution in advance and give students more freedom to choose abstract concepts for creating a solution. As a result, students create their own animations games or mobile applications to solve real-world problems providing opportunities for students “to extend their creative expression to solve problems, create computational artifacts” (Yadav and Cooper, 2017).

3. Definition and Execution of the Systematic Mapping Study

In order to elicit the state of the art on approaches for assessing product creativity (or some of its characteristics) based on the analysis of computer programs developed by students in an educational context as an outcome of the learning process, we performed a systematic mapping following the procedure defined by Petersen *et al.* (2015).

3.1. Definition of the Review Protocol

Research Question. Which studies exist for the assessment of product creativity, or some of its characteristics, of computer programs in the educational context?

We refined this research question into the following analysis questions:

AQ1. Which studies exist and for what kind of product and educational stage?

AQ2. What is the definition of the product creativity characteristics being assessed?

AQ3. How are these creativity characteristics analyzed?

AQ4. What is the context and sample size of the application of the approach?

AQ5. If, and how the approach has been evaluated?

Data source. We examined all published English-language articles that are available on Scopus, the largest abstract and citation database of peer-reviewed literature, including publications from ACM, Elsevier, IEEE and Springer with free access through the Capes Portal¹.

Inclusion/exclusion criteria. We considered only peer-reviewed English-language articles that present a form of assessment of product creativity, or some of its characteristics, based on the analysis of computer programs. Here we understand assessment to a wide extent, including any kind of measurement or analysis of the creative product in an educational context, considering also approaches that not necessarily

¹ A web portal for access to scientific knowledge worldwide, managed by the Brazilian Ministry on Education for authorized institutions, including universities, government agencies and private companies (www.periodicos.capes.gov.br).

include grading and/or a feedback mechanism. We only consider studies within an educational context. Due to the sparseness of research in this area, we include approaches to all educational stages. We consider articles that have been published until December 2019.

On the other hand, we exclude studies that assess creativity only related to other strands of creativity than the product. Thus, studies focusing exclusively on press, person, and process creativity are excluded. We consider only articles that present substantial information to enable the extraction of relevant information regarding the analysis questions, and, therefore, abstract-only or one-page articles are excluded.

Definition of the search string. Following our research objective, we define the search string by identifying core concepts, also considering synonyms, as indicated in Table 3. The term creativity is chosen, as it expresses the main concept to be searched. As originality is one characteristic widely accepted in the field (Runco and Albert, 2010) we choose to use it as a synonym of creativity to broaden the search results. Although originality alone is not sufficient to classify a product as being creative, independently of what other positive qualities it may have, it is generally considered an important characteristic for a creative product to possess (Jackson and Messick, 1964). We do not use other synonyms of originality such as novel, infrequency or unusualness, as these terms are used in many contexts with meanings unrelated to creativity. We also focus the search on the keyword assessment, including synonyms that are commonly used in the educational context. Keywords related to educational context are chosen to limit results to this specific context. Considering our focus on computing education, specifically the assessment of product creativity based on computer programs, we also include terms related to this domain. We use computational thinking as a synonym for programming/coding. And, although computational thinking covers a much wider field than just programming and coding, it is frequently used as a synonym to these terms in the literature (Armoni, 2016). We use wildcard characters to cover as many variations of the terms as possible, such as *creativ** representing “creative” and “creativity”.

Using these keywords, the search string has been calibrated and adapted in conformance with the specific syntax of the data source:

TITLE-ABS-KEY((*creativ** OR *original**) AND (*assess** OR *measur** OR *evaluat** OR *analy**) AND (“K-12” OR school OR education OR learning) AND (coding OR programming OR “computational thinking”))

Table 3
Keywords

Concept	Keywords and synonyms
Creativity	<i>creativ*</i> ; <i>original*</i>
Assessment	<i>assess*</i> ; <i>measur*</i> ; <i>evaluat*</i> ; <i>analy*</i>
Educational context	K-12; school; education; learning
Programming artifact oriented	Programming; coding; computational thinking

3.2. Execution of the Search

The search has been executed in January 2020 by the first author and revised by the co-authors. As Scopus allows to filter results based on the field, we choose to exclude works on unrelated fields such as Medicine (225), Biochemistry, Genetics, and Molecular Biology (185), Agricultural, Biological Sciences (158), Business, Management and Accounting (156), Health Professions (34), Nursing (26), Economics, Econometrics, and Finance (24), Pharmacology, Toxicology, and Pharmaceutics (13), Immunology and Microbiology (12), Dentistry (2), and Veterinary (1). In the first analysis stage, we quickly reviewed titles, abstracts, and keywords of all filtered search results (1,837 articles) to identify articles that match the exclusion criteria, resulting in 89 potentially relevant articles. In the second stage, we analyzed the full-text of the pre-selected articles. We found 11 articles that analyze product creativity based on computer programs created by the students. All authors participated in the selection process and discussed the selection of papers until a consensus was reached (Table 4).

Many articles have been excluded based on the analysis of their abstracts as they are related to other fields such as video coding, artificial intelligence, deep learning, because 'original' is a term widely used to describe, for example, datasets. In addition, a large number of articles using the term 'originality' with a different meaning only to indicate the novelty of a study have been excluded. During the full-text analysis of the remaining

Table 4
Quantity of articles per selection stage

Initial search results	Filtered search results (Excluding works in unrelated fields)	Potentially relevant articles selected based on the analysis of title and abstract	Selected articles based on full-text analysis
2,473	1,837	89	11

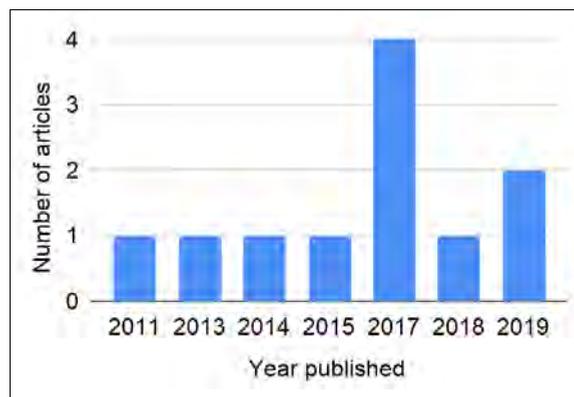


Fig. 2. Amount of articles presenting an assessment of product creativity based on programming artifacts in the educational context per year.

Table 5
Articles selected

ID	Title	Reference
1	Computing indicators of creativity	Koh <i>et al.</i> , 2011
2	Computing creativity: Divergence in computational thinking	Bennett <i>et al.</i> , 2013
3	Automated indicators to assess the creativity of solutions to programming exercises	Manske & Hoppe, 2014
4	An exploration of three-dimensional integrated assessment for computational thinking	Zhong <i>et al.</i> , 2015
5	Suggesting a log-based creativity measurement for online programming learning environment	Gal <i>et al.</i> , 2017
6	Identifying original projects in app inventor	Mustafaraj <i>et al.</i> , 2017
7	Computational thinking development through creative programming in higher education	Romero <i>et al.</i> , 2017
8	Work in progress: Identifying and analyzing original projects in an open-ended blocks programming environment	Turbak <i>et al.</i> , 2017
9	What we can learn about student learning from open-ended programming projects in middle school computer science	Grover <i>et al.</i> , 2018
10	Using Rubrics Integrating Design and Coding to Assess Middle School Students' Open-ended Block-based Programming Projects	Basu, 2019
11	Creativity in the acquisition of computational thinking	Hershkovitz, 2019

articles, several have been excluded as they present approaches exclusively analyzing creativity concerning other strands that are outside of the focus of our research, such as press (Engelman *et al.*, 2017), process (Perez-Poch *et al.*, 2016) or person (Engelman *et al.*, 2017). As a result, we identified a total of 11 articles relevant to our research objective (Table 5). All selected articles were published within the last nine years as shown in Fig. 2, which also indicates the recent importance of this topic.

As Koh *et al.* (2011) and Bennett *et al.* (2013) present the same approach: Computational Thinking Patterns (CTP) and creativity analysis through Computational Thinking Pattern Analysis (CTPA) (Koh *et al.*, 2011; Bennett *et al.*, 2013), both articles are grouped in the analysis.

4. Data Analysis

In this section, we present our findings for each of the analysis questions.

4.1. Which studies exist and for what kind of product and educational stage?

We found 11 articles describing 10 approaches, as two articles present the same approach just from a different perspective (Koh *et al.*, 2011; Bennett *et al.*, 2013). The approaches purposefully analyze a student's program created as an outcome of the learning process to assess creativity, identify characteristics of product creativity, and study its relationship

with other concepts, e.g., computational thinking. In the computing education context, the products assessed vary from games (Koh *et al.*, 2011), solutions to well-defined programming activities (Manske and Hoppe, 2014; Gal *et al.*, 2017), mobile apps developed in class and published to public galleries (Mustafaraj *et al.*, 2017; Turbak *et al.*, 2017), projects as results of creative programming activities (Romero *et al.*, 2017) as well as free-choice, open-ended projects (Grover *et al.*, 2018; Basu, 2019). Some products are a result of a well-defined activity (with a known solution in advance) or ill-defined activity (without or with more than one known solution known in advance) (Table 6).

The analysis of creativity based on the students' computer programs is provided for diverse programming environments/languages, especially block-based visual programming environments such as Scratch and App Inventor that are typically used for computing education in K-12 (Fig. 3). Some studies are also conducted in a more generic manner covering more than one programming environment/language, e.g., using rubrics proposed by Basu (2019) or Grover *et al.* (2018).

The approaches target different educational stages. The majority of studies were designed for some stage in K-12 education (Bennett *et al.*, 2013; Basu, 2019; Grover *et al.*,

Table 6
Products and activities for which they are designed

Product	Type of educational activity		Examples
	Open-ended well-structured	Open-ended ill-defined	
Games, histories, etc.		x	Grover <i>et al.</i> , 2018; Basu, 2019; Romero <i>et al.</i> , 2017
Game	x		Koh <i>et al.</i> , 2011; Bennett <i>et al.</i> , 2013
Solutions to online activities	x		Manske & Hoppe, 2014; Hershkovitz, 2019; Gal <i>et al.</i> , 2017
Creative design report		x	Zhong <i>et al.</i> , 2015
Mobile apps		x	Turbak <i>et al.</i> , 2017; Mustafaraj <i>et al.</i> , 2017

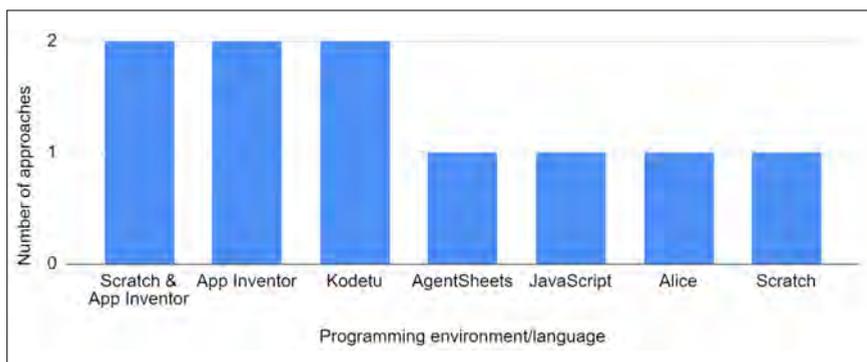


Fig. 3. Amount of approaches per programming environment.

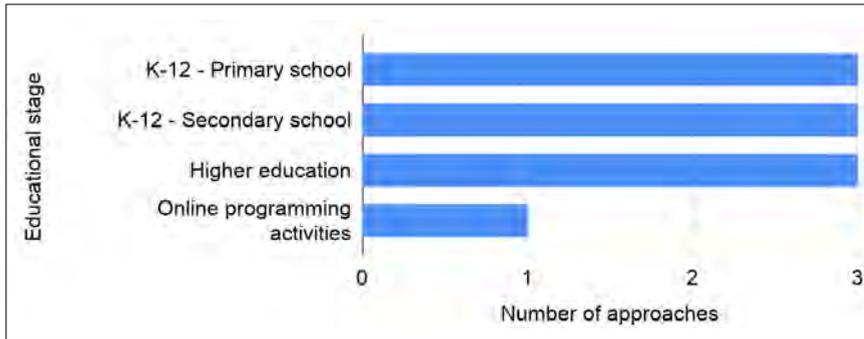


Fig. 4. Amount of approaches per educational stage.

2018), while others target higher education (Romero *et al.*, 2017; Mustafaraj *et al.*, 2017; Turbak *et al.*, 2017) (Fig. 4). One approach uses as input data from a website that contains a set of well-defined programming activities (Manske and Hoppe, 2014) without clearly indicating the considered educational stage.

4.2. What is the Definition of the Product Creativity Characteristics Being Assessed?

In alignment with the specific focus of this review, all articles focus on product creativity only, except for the work by Hershkovitz *et al.* (2019), which also analyzed characteristics related to the creative process. One article does not inform which characteristics of creativity were analyzed (Romero *et al.*, 2017).

Detailing the specific characteristics of product creativity that are assessed, we observe that the most analyzed characteristic is originality, analyzing the newness of the product. And, although, authors use different terms, such as novelty (Basu, 2019; Grover *et al.*, 2018) or divergence (Bennett *et al.*, 2013; Koh *et al.*, 2011), they refer to the same concept of originality. Grouping these terms, originality was analyzed by 8 of 10 approaches that informed the analyzed characteristics (Fig. 5).

Typically, originality is assessed by comparing the student's computer program with a specific set of computer programs. This set can contain all other student's computer programs (Gal *et al.*, 2017), or pre-programmed solutions and patterns for well-defined activities (Koh *et al.*, 2011). The indicator of originality, novelty or divergence is then measured by the extent to which the student's computer program is different from this set. Originality is also assessed by using more subjective criteria, as proposed by Grover *et al.* (2018) and Basu (2019), having the instructor to assess computer programs on an ordinal scale as "not very novel, some novelty, or very novel" (Basu, 2019).

The characteristic condensation seems to be fuzzier to assess than originality, as it depends on the domain-specific interpretation (Manske and Hoppe, 2014). Different to product originality that can be somewhat agreed on independently from the domain, terms related to condensation, such as sophistication and elegance are very domain-

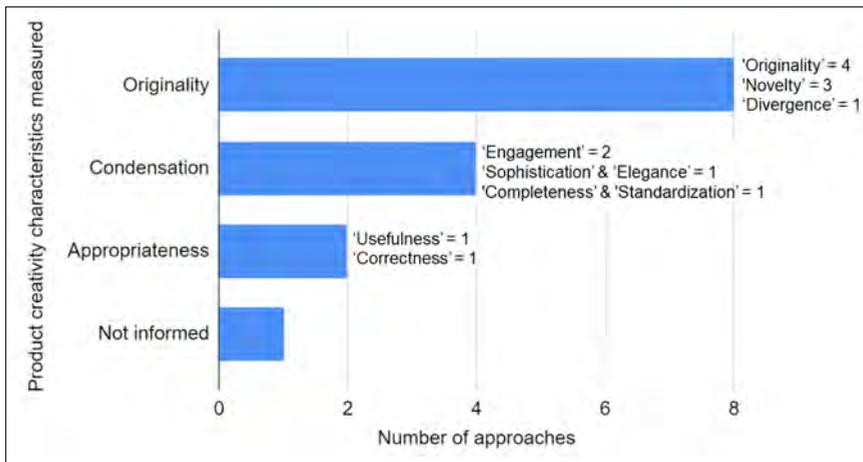


Fig. 5. Creativity product characteristics analyzed in the selected studies.

specific without a general agreement outside specific domains. In this regard, in the context of software engineering, elegance is measured using software metrics where “experts infer a weighting and interpretation to these metrics” (Manske and Hoppe, 2014). Other terms related to condensation, such as completeness and standardization are more straightforward to assess. For example, completeness is measured by verifying if the computer program is completed and standardization if the computer program follows some defined pattern or formatting rule (Zhong *et al.*, 2015).

Appropriateness is assessed only by a few approaches. Manske and Hoppe (2014) use the term usefulness and defined that it is achieved if the student’s computer program is correct for the activity for which it was submitted (Manske and Hoppe, 2014). Basu (2019) uses the term correctness in a similar way and defines more subjective assessment criteria from “the programs contain several errors” to “program runs correctly without error and the output is appropriate”.

4.3. How are these Creativity Characteristics Analyzed?

The approaches encountered in this mapping vary largely concerning the type of assessment and methods and techniques used. With respect to whom performs the assessment, we found instructor assessment using rubrics (Grover *et al.*, 2018; Basu, 2019), expert assessment based on personal knowledge as input to automated assessment (Manske and Hoppe, 2014) as well as automated assessment based on techniques from computer science and mathematics (Koh *et al.*, 2011; Bennett *et al.*, 2013; Gal *et al.*, 2017; Mustafaraj *et al.*, 2017; Turbak *et al.*, 2017). Studies that focus on the relationship between creativity and computational thinking analyze characteristics of the process rather than the product, using psychological tests, such as the Torrance Tests of Creative Thinking (Hershkovitz *et al.*, 2019). Half the approaches propose assessment techniques that are performed manually, while the other half are automated.

One way of assessing the creative product is by the instructor manually assessing the outcome created by the students using rubrics (Fig. 6). Rubrics consists of a matrix of criteria and performance levels for each criterion. Such rubrics typically contain one or more criteria related to creativity or its characteristics (Fig. 6), e.g., novelty or originality and condensation or engagement, along with the respective performance levels (Grover *et al.*, 2018; Basu, 2019; Zhong *et al.*, 2015).

Using concepts of education, computer science, and mathematics, the approaches analyze product creativity. In addition, the study presented by Hershkovitz *et al.* (2019) also uses the TTCT test as a psychological instrument (Fig. 7) for measuring creativity concerning the process strand.

Several approaches visioning the automation of the assessment adopt metrics with machine learning models to assess creativity (Manske and Hoppe, 2014) or to identify original projects (Turbak *et al.*, 2017; Mustafaraj *et al.*, 2017). Some apply machine learning models, including regression methods, such as linear regression and support vector regression (Manske and Hoppe, 2014), clustering methods, such as the Markov cluster algorithm (Turbak *et al.*, 2017) and the K-nearest neighbors' algorithm (Mustafaraj *et al.*, 2017). The input for these algorithms included features gathered using statistical concepts, such as term frequency-inverse document frequency (TF-IDF) (Turbak *et al.*, 2017) and the Jaccard index of similarity (Mustafaraj *et al.*, 2017). Some features were defined using software engineering metrics, such as effective lines of code, visited lines of code, and cyclomatic complexity (Manske and Hoppe, 2014). Abstract language tokens (obtained during the lexical analysis phase) were also used for comparing dis-

OVERALL PROFICIENCY				
	Lacking basic proficiency 0 point	Emerging proficiency 1 point	Proficient 2 points	Exceeding grade level proficiency 3 points
Program Correctness	Programs contains several errors	Program contains a few errors	Program runs without errors, but the functionality and/or output is inappropriate (semantic errors)	Program runs correctly without errors and the output is appropriate
Code Readability	There are no comments or user instruction for running the code	The source code is somewhat readable; it is partly commented or has some partial user instruction	The source code is either fully commented or has complete user instruction	The source code is fully commented and has complete user instruction
Novelty	Not meeting minimal requirements	Not very novel	Some novelty	Very novel
Engagement	Not meeting minimal requirements	Not very engaging	Somewhat engaging	Very engaging
Complexity	Not meeting minimal requirements	Not very complex	Somewhat complex	Very complex

Fig. 6. Example excerpt of the rubric proposed by Basu (2019).

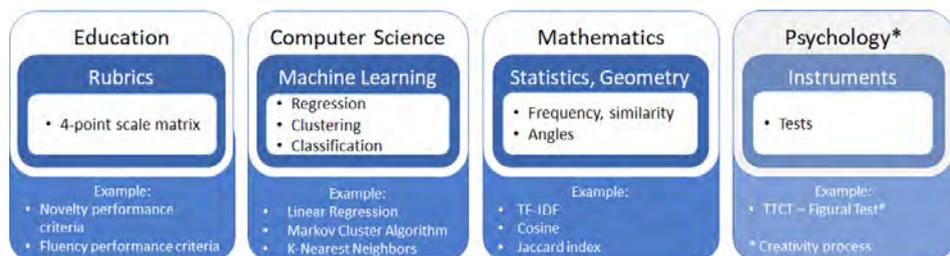


Fig. 7. Concepts used to analyze creativity product characteristics in the selected articles (* related to the process strand).

tances between artifacts using string metrics (Manske and Hoppe, 2014). In general, supervised learning methods were used to train machine learning models to assess creativity (Manske and Hoppe, 2014), while unsupervised learning methods were adopted for the identification of originality characteristic in projects (Turbak *et al.*, 2017; Mustafaraj *et al.*, 2017).

Studying the relationship between computational creativity and computational thinking, Hershkovitz *et al.* (2019) use the Torrance Tests of Creative Thinking (TTCT) – Figural Test to capture the creative process. The output from the TTCT is then compared with the statistical infrequency of products, or solutions to programming exercises, created by the students. The Computational Thinking Pattern Analysis (CTPA) uses mathematical concepts to analyze creativity divergence in the student’s programming solutions compared to patterns previously defined (Bennett *et al.*, 2013; Koh *et al.*, 2011). The authors used the cosine to provide a result regarding the difference between the tutorial and the student’s computer program.

Regarding scalability, automation, and robustness, some approaches also use automated analysis such as the CTPA divergence analysis (Bennett *et al.*, 2013; Koh *et al.*,

Table 7
Assessment techniques per product creativity characteristic

Characteristic	Assessment technique	Manual (M) or Automated (A) approach	Example
Originality	Education (rubric criteria)	M	Novelty criterion – 0 point: Not meeting minimal requirements, 1 point: Not very novel, 2 points: Some novelty, 3 points: very novel (Basu, 2019)
	Computer science	A	K-nearest neighbors algorithm with Euclidean distance (Mustafaraj <i>et al.</i> , 2017)
	Mathematics	A	Similarity calculated using cosine (Bennett <i>et al.</i> , 2013) and infrequency of solution among all solutions (Hershkovitz <i>et al.</i> , 2019)
Condensation	Education (rubric criteria)	M	Engagement criterion – 0 point: Not meeting minimal requirements, 1 point: Not very engaging, 2 points: Somewhat engaging, 3 points: very engaging (Basu, 2019)
	Computer science	A	Software engineering metrics to measure elegance (Manske and Hoppe, 2014)
	Mathematics	A	Not found
Appropriateness	Education (rubric criteria)	M	Correctness criterion – 0 point: Programs contains several errors, 1 point: Program contains a few errors, 2 points: Program runs without errors, but the functionality and/or output is inappropriate (semantic errors), 3 points: Program runs correctly without errors and the output is appropriate (Basu, 2019)
	Computer science	A	Usefulness is achieved by comparing the output with the desired output (Manske and Hoppe, 2014).
	Mathematics	A	Not found

Table 8
Assessment strategies for providing scores

Assessment strategy	Reference
Math formula for assessing a score on <i>divergence (originality)</i>	Bennett <i>et al.</i> , 2013; Koh <i>et al.</i> , 2011
4-point rating scale for assessing scores on <i>product creativity characteristics</i>	Grover <i>et al.</i> , 2018; Basu, 2019
Thresholds for rarity or complementary to 100% of the frequency of the solution among all the correct solutions for assessing a score on <i>originality</i>	Hershkovitz <i>et al.</i> , 2019; Gal <i>et al.</i> , 2017
Labeling as original or unoriginal using Machine Learning models for assessing <i>originality</i>	Mustafaraj <i>et al.</i> , 2017; Turbak <i>et al.</i> , 2017
7-star rating scale by experts and as an input in Machine Learning models for assessing a score on <i>product creativity</i>	Manske and Hoppe, 2014
8 points for assessing scores on <i>product creativity characteristics</i>	Romero <i>et al.</i> , 2017
5-point rating scale for assessing scores on <i>product creativity characteristics</i>	Zhong <i>et al.</i> , 2015

2011) or the identification of originality in projects (Turbak *et al.*, 2017; Mustafaraj *et al.*, 2017). The automated analysis allows anyone (including the students themselves) to assess the products quickly and receive instant feedback on their performance (Table 7).

Regarding instructional feedback and grading, the approaches typically calculate a score for the student's computer program. Depending on the characteristic being assessed, some approaches use rating scales with performance levels specifying more complex product characteristics as the score increases. Others, such as Koh *et al.* (2011) Bennett *et al.* (2013), calculate the scores mathematically (Table 8). On the one hand, approaches adopting machine learning classification models provide a result that indicates if the computer program was classified as original or unoriginal, without assigning a score. On the other hand, machine learning regression models provide a score based on the expert rating scale in the datasets (Table 8). None of the approaches presents instructional feedback such as tips or suggestions to constructively guide the learning process based on the assessment results.

4.4. What is the Context and Sample Size of the Application of the Approach?

The majority of the approaches were applied in face-to-face K-12 school classes (Fig. 8), such as a middle school in a large urban school district in the Western US (Grover *et al.*, 2018), a primary school in Spain (Hershkovitz *et al.*, 2019), and primary school in Changshu City of China (Zhong *et al.*, 2015). Blended applications with face-to-face and online classes were applied in the middle school context (Bennett *et al.*, 2013; Koh *et al.*, 2011). Face-to-face university classes included a course at Wellesley College in the US (Mustafaraj *et al.*, 2017) and the Université Laval in Canada (Romero *et al.*, 2017).

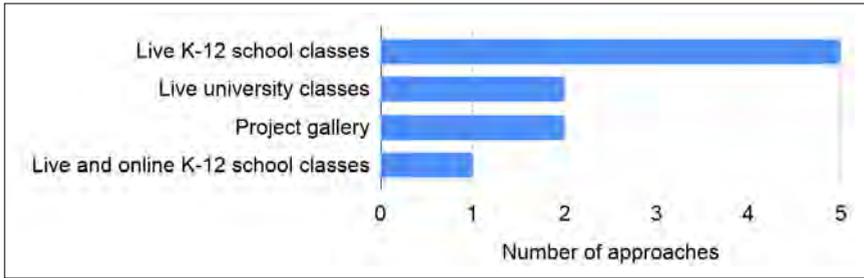


Fig. 8. Context of data (application) of the works presented in the selected articles.

Table 9
Sample size variations per educational stage

Context	Subjects	Sample analyzed
Face-to-face K-12 school classes	~80 to ~214 students	~80 to ~1332 projects
Face-to-face and online K-12 school classes	~296 students (inferred from projects analyzed)	~296 projects
Face-to-face university classes	~16 to ~120 students	~120 to 902 projects
Project galleries	~6000 to ~260717 users (inferred)	~4099877 projects (inferred)

Some of the approaches have also been evaluated by using projects from repositories providing support for the sharing of projects among students, from which solutions created by students can be downloaded and analyzed. Here, specifically, the App Inventor Gallery was used (Turbak *et al.*, 2017) as well as the Project Euler (Manske and Hoppe, 2014) to obtain thousands of students' projects.

The sample size varies from small samples in the university context (Mustafaraj *et al.*, 2017) and face-to-face classes (Hershkovitz *et al.*, 2019) to large samples obtained from online project galleries (Turbak *et al.*, 2017; Manske and Hoppe, 2014). Some values were inferred based on the numbers provided by the authors, for example, Manske and Hoppe indicate that Project Euler (online gallery) had 4099877 solutions in November 2012, yet do not explicitly state if all solutions were used for the analysis. Therefore, we assumed that the sample size is equal to the number of projects by indicating this as an inferred value in Table 9.

4.5. If, and How the Approach Has Been Evaluated?

Most articles do not present an evaluation of the approach, as this may have been outside the scope of the articles. Furthermore, Hershkovitz *et al.* (2019) assume the reliability and validity of the psychological test (TTCT) as the test has been widely evaluated beforehand.

Exceptions are Basu (2019) for the evaluation of the defined rubric as well as Manske and Hoppe (2014) and Mustafaraj *et al.* (2017) related to the machine-learning-

based approaches proposed. Basu (2019) used Cohen's kappa coefficient to analyze the inter-rater reliability of the proposed rubric. A value of 0.9 was found by scoring discrepancies among teacher's scores. After computing the coefficient, teachers also scored additional projects independently providing additional opportunity to refine the rubric based on their feedback.

Manske and Hoppe (2014) performed a reliability evaluation on the agreement of expert assessments, which were used as an input to the proposed machine learning model. Yet, the evaluation of the machine learning model did not provide meaningful results due to the lack of agreement between raters. As raters used individual creativity definitions in a not consistent way, this resulted in two different groups of agreement measured via Krippendorff's alpha coefficient, with low values (below 0.3) indicating no agreement between raters. However, they found a high agreement within the specific theorists-group from the educational context (Krippendorff's coefficient 0.729) and medium agreement within software engineering related experts from industry (Krippendorff's coefficient 0.552), indicating that the two groups can be separated in terms of assessing product creativity.

Mustafaraj *et al.* (2017) analyzed the accuracy of the classification of original and unoriginal projects regarding the Jaccard distance. They found an accuracy of 89% for both classes using a 0.4 Jaccard distance. A value less or greater than this results in diminishing the accuracy from one class, thus labeling it wrong, e.g. a more than 11% original projects may be labeled unoriginal if the value for distance is not 0.4. This result is particularly related to classification approaches for determining proper thresholds.

Some authors also mention some sort of evaluation of the proposed approaches without providing further details. Grover *et al.* (2018) report joint discussions about a few projects to establish interrater reliability yet without providing further information. Koh *et al.* (2011) argued that the divergence calculation used in their approach is supported by other data sources and that the validity of the approach demonstrates uniqueness in three separate learning conditions.

Hershkovitz *et al.* (2019) compared the results from the TTCT test for creative thinking with the assessment of originality of the students' computer programs to well-defined problems. They found significant correlations between the two types of creativity measures and in some cases "creativity in programming is positively associated with the broad construct of creativity" (Hershkovitz *et al.*, 2019).

5. Discussion

Considering the importance of creativity as a 21st-century skill, only very few assessment approaches have been encountered in the context of computing education with active learning strategies for assessing the student's creative product. And, although there exist already a considerable number of approaches for assessing computing education in general, these mostly focus exclusively on computational thinking concepts and practices (Moreno-León and Robles, 2015; Gresse von Wangenheim *et al.*, 2018). Only very few of them also include assessment criteria related to creativity on the strand of

the product (Koh *et al.*, 2011; Basu 2019), some of them in a superficial way as one subjective criterion to be judged manually by the instructor or peer. Popular automated assessment tools for assessing outcomes of computing education created with block-based programming languages such as Dr. Scratch (Moreno-León and Robles, 2015) or CodeMaster (Gresse von Wangenheim *et al.*, 2018) also do not assess creativity.

Analyzing specifically approaches for assessing product creativity of computer programs created by students as part of computing education, it becomes clear that the definition of creativity strongly influences how approaches assess the product. Considering that the term depends on many variables, it is not sufficient to only define which strand is being analyzed. This issue is further complicated through context-dependency as characteristics analyzed can vary as well have different meanings for the same terms in different contexts, for example, the usefulness of an app can be understood as if the app allows the user to perform the desired tasks, while related to a game it can be seen as if the game is pleasurable to play. In this context, it seems that originality is one of the few well-established characteristics in the literature regarding product creativity.

Most of the approaches are not based on well-known product creativity assessment models and simply use the term with a meaning of general common sense without presenting an in-depth analysis of the field for defining what comprises the assessment of product creativity. Some of the approaches focus on one product characteristic as a way of assessing creativity, excluding other characteristics typically considered of general product creativity assessment. As we also included originality in the search string, some approaches that focus exclusively on originality (not on the “creativity” assessment) do not assess creativity itself and aim at assessing originality as a singular construct, which may not provide an in-depth analysis of creativity. Approaches that assess condensation include those that define the term using well-known models as well as others defining the concept only superficially. Terms used for condensation are completeness, elegance and sophistication or engagement. Appropriateness is defined only by two approaches and, again, one of them uses well-known models for its definition, while the other defines it superficially. The two terms related to appropriateness are correctness and usefulness.

As the definition of creativity also depends on the specific context, the existing approaches tailor well-known product creativity assessment characteristics to the context of computing education. Thus, the definition of each characteristic assessed is related to computing concepts. Originality is typically customized by comparing the students’ computer programs to identify the frequency of different solutions. Condensation is customized as a well-designed interface usability or software engineering metrics regarding the software architecture of the computer program (Manske and Hoppe, 2014). Appropriateness is measured by comparing the output of the students’ computer program with the desired output for well-defined activities (Manske and Hoppe, 2014). However, for ill-defined activities so far there are only manual approaches that tailor creativity assessment to assess usefulness depending on subjective criteria such as if the program runs with many/few errors.

Only a few approaches explicitly define customizations, yet, lacking the indication of a theoretic background for several definitions. This shows that it is imperative to move towards a more precise definition of creativity in the context of computing education.

This would provide a shared understanding of the construct as a basis for the design and development of reliable and valid assessments.

Another issue is related to the object being assessed. In the context of product assessments, the approaches encountered are based on the assessment of a single product (in this case, computer programs). However, as typically creating a product includes also creating other intermediate outcomes, it may be important to consider not only the single end result of the learning process for the assessment but also these intermediate outcomes, such as requirements specifications, the interface design and/or test cases in the context of software development.

Half of the approaches encountered propose manual assessments, yet, these may be biased and time-intensive to complete. This becomes especially problematic in the context of large classes or Massive Open Online Courses (MOOCs). Even if manual assessments provide a context tailored result, it may be impossible to provide constant timely feedback throughout the learning process. And, although such manual assessments can also rely on peers with results that align with instructor assessments reducing the instructor's assessment effort, they are still subjective and require substantial time and organization (Miller *et al.*, 2014). These reasons may limit the utility of manual approaches as the sole assessment alternative.

Yet, as creativity is complex and multi-dimensional it can be expressed in diverse ways, just an automated assessment of computer programs may not be sufficient as a single way to account for all its facets. Product-oriented approaches to the assessment of creativity are sometimes criticized for under-representing the creativity of individuals (Couger and Dangate, 1996). Thus, in order to capture creativity in a more comprehensive way it may be beneficial to adopt diverse approaches, e.g., completing an automated objective assessment of the product, through the manual subjective assessment by peers and/or instructors. However, none of the approaches we encountered suggests such a strategy. Only one approach compares the results of product assessment with the results of the TTCT test, yet, to study the relationship between creativity and computational thinking rather than to provide a holistic assessment of creativity.

In order to properly assess the concept, it is essential to provide a robust assessment model. Considering the complexity of the concept of product creativity and the lack of well-established definitions, a further shortcoming observed is the lack of more scientifically rigorous evaluations concerning the reliability and validity of the proposed assessment approaches. And, although several studies are based on considerable samples, typically using artifacts from product sharing platforms, such as App Inventor Gallery and Project Euler, these may not provide detailed context information. Thus, it is not possible to analyze these approaches for specific educational stages as the data comes from unknown origins. Therefore, the data of these studies may not be representative of the specific educational stage nor the specific target population with whom the approach will be used.

Given that assessors are critical in manual assessments, understanding who they are and what their level of expertise is, is also important as it has a direct impact on inter-rater agreement and reliability. As the human judgment of creativity remains by nature an intrinsically subjective process, it is necessary to study to which degree the perception

of creativity of computer programs is consistent and not being an idiosyncratic result of an assessor's subjective judgment. In this regard, Manske and Hoppe (2014) found that assessors with an educational theoretical background are more likely to provide a consistent assessment of product creativity in the context of computing education. Taking into consideration that currently teachers formally trained in computer science are scarce in K-12, as well as self- or peer-assessment conducted by students still learning computing, this question has to be considered carefully in the design of the assessment instruments. This is particularly important in order to assure consistency regarding the meaning of assessment criteria and the performance levels by the assessors.

In general, the approaches only indicate as a result of the assessment a performance level typically on an ordinal scale. These scales are developed using Classical Test Theory, representing creativity of the product as the sum of the score, e.g., in Zhong *et al.* (2015). Alternatives such as a definition of a scale based on Item Response Theory may be a more appropriate way of creating a construct for assessing product creativity. However, none of the encountered approaches uses Item Response Theory, although Myszkowski and Storme (2019) argue that Item Response Theory-based scoring can lead to a more appropriate and accurate estimation of the latent trait (the creative value of the product), questioning also common practices regarding the aggregation of ratings.

The results of our analysis also point out a lack of the provision of more comprehensive instructional feedback. Except for a grade, no additional instructional feedback is given to the students based on the assessment results to guide the learning process. Such feedback, given constructively is important for the student to understand what are the strengths and weaknesses of the product, and consequently learning opportunities, as well as to the instructor to improve retention and knowledge transfer, e.g. a comprehensive explanation of why a student product is considered unoriginal. Most of the approaches also do not propose how to use these results as part of a summative assessment for grading. For example, Manske and Hoppe (2014) propose to use the score to classify the student's computer program. Different to other automated approaches in the context of computing education such as Dr. Scratch (Moreno-León and Robles, 2015) or CodeMaster (Gresse von Wangenheim *et al.*, 2018), none of the approaches uses any kind of ludic representation of the results of the assessments (such as badges, ninja belts, etc.) to motivate students, especially in K-12.

Another issue that seems not to be considered is to which respect the assessment itself may inhibit creativity in schools, as learning experiences that involve comparisons to others, emphasis on extrinsic features of the task, and the pressure of being evaluated may cause anxiety and impair motivation and capacity for creativity (Runco, 2003). Thus, these impacts of the assessment also need to be studied and taken into consideration when designing the assessment to minimize their consequences.

Considering that creativity is a central competence of the 21st century, the lack of wider research on the assessment of learning creativity on the strand of the product as part of computing education is surprising. Although the study of creativity in computing education on the strand of the process seems to be more approached, this indicates a need for future work in the area of the product to effectively and efficiently support the teaching and learning of creativity as part of computing education in practice.

The availability of reliable and valid approaches is also essential to systematically create a body of empirical evidence supporting the assumption that computing education also contributes to the development of creativity, especially on the strand of the product, as systematic research on this issue is still scarce mostly dating back to the 1980s and 1990s (Clements, 1995).

Threats to Validity. Systematic mappings may suffer from the omission of relevant studies. In order to mitigate this risk, we carefully constructed the search string to be as inclusive as possible, considering not only core concepts but also synonyms. We also included originality as one of the characteristics widely present for product creativity to include works analyzing this specific characteristic. We also searched multiple databases indexed by Scopus, which covers the majority of scientific publications in this field. Threats to the study selection and data extraction were mitigated by providing a detailed definition of the inclusion and exclusion criteria. We defined and documented a rigid protocol for the study selection and the selection was conducted by all co-authors together until consensus was achieved. The lack of a clear definition of product creativity in the context of computing education was also mitigated using a set of keywords related to the characteristics of general product creativity. And, although we found only 11 articles describing 10 approaches, the overview presented here shows an in-depth analysis of important aspects regarding the assessment of creativity proposed by the approaches. Data extraction was inferred in some cases, as the relevant information was not always explicitly reported. In these cases, the inference made by the first author and carefully reviewed by the co-authors was indicated throughout the article.

6. Conclusions

From this review, it becomes evident that despite a current trend towards the teaching and learning of creativity in K-12, approaches for the assessment of product creativity in the context of computing education are just emerging. We only encountered 10 relevant approaches aiming at the assessment of computer programs created mostly with block-based programming languages, such as Scratch and App Inventor typically used in K-12 as well as few targeting higher education. These revealed a lack of a commonly accepted definition of product creativity customized to the context of computer education, confirming only originality as one of the well-established characteristics. Furthermore, several approaches seem to lack clearly defined criteria for effective, efficient and useful creativity assessment, especially in K-12. Diverse techniques are used including rubrics, mathematical models as well as machine learning, supporting manual as well as automated approaches. However, very few performed an evaluation analysis of the proposed approach, thus not providing results indicating reliability and validity. These results indicate the need for further research to support the assessment of product creativity in the context of computing education in a more effective, efficient way that can easily be adopted in educational practice as well as the need for more robust (systematic defined and validated) assessment models for creating an empirical basis to study the development of creativity in K-12 educational contexts.

Acknowledgments

This study was financed in part by the *Coordenação de Aperfeiçoamento de Pessoal de Nível Superior*–Brasil (CAPES)–Finance Code 001 and by the *Conselho Nacional de Desenvolvimento Científico e Tecnológico* – Brasil (CNPq) – Grant No.: 302149/2016–3.

References

- Aldave, A., Vara, J.M., Granada, D., Marcos, E. (2019). Leveraging creativity in requirements elicitation within agile software development: A systematic literature. *Journal of Systems and Software*, 157.
- Alves, N. da C., Gresse von Wangenheim, C., Hauck, J. C. R. (2019). Approaches to Assess Computational Thinking Competences Based on Code Analysis in K-12 Education: A Systematic Mapping Study. *Informatics in Education*, 18(1), 17–37.
- Amabile, T.M. (1982). Social psychology of creativity: A Consensual Assessment Technique. *Journal of Personality and Social Psychology*, 43, 997–1013.
- Araujo, A.L.S.O. de, Andrade, W.L., Serey Guerrero, D.D. (2016). A systematic mapping study on assessing computational thinking abilities. In *Proc. of the IEEE Frontiers in Education Conference*, 1–9. IEEE.
- Armoni, M. (2016). Computer science, computational thinking, programming, coding: the anomalies of transitivity in K-12 computer science education. *ACM Inroads*, 7(4), 24–27.
- Basu, S. (2019). Using Rubrics Integrating Design and Coding to Assess Middle School Students' Open-ended Block-based Programming Projects. In *Proc. of the 50th ACM Technical Symposium on Computer Science Education*, 1211–1217. ACM.
- Beghetto, R.A. (2010). Creativity in the classroom. In J. C. Kaufman, & R. J. Sternberg (Eds.), *Cambridge handbook of creativity*, 447–463. New York, NY: Cambridge University Press.
- Bennett, V.E., Koh, K.H., Repenning, A. (2011). A. CS education re-kindles creativity in public schools. In *Proc. of the 16th Annual Joint Conference on Innovation and Technology in Computer Science Education*, 183–187. ACM.
- Bennett, V.E., Koh, K.H., Repenning, A. (2013). Computing creativity: divergence in computational thinking. In *Proc. of the 44th ACM Technical Symposium on Computer Science Education*, 359–364. ACM.
- Besemer, S. P. (2000). Creative Product Analysis to Foster Innovation. *Design Management Journal* (Former Series), 11, 59–64.
- Besemer, S., Treffinger, D.J. (1981). Analysis of Creative Products: Review and Synthesis. *Journal of Creative Behavior*, 15(3), 158–178.
- Bialik, M., Martin, J., Mayo, M., & Trilling, B. (2016). Evolving Assessments for a 21st Century Education. *Center for Curriculum Redesign*.
- Bjögvinsson, E., Ehn, P., Hillgren, P.-A. (2012). Design Things and Design Thinking: Contemporary Participatory Design Challenges. *Design Issues*, 28(3), 101–116.
- Bolden, B., DeLuca, C., Kukkonen, T., Roy, S. Wearing, J. (2019). Assessment of Creativity in K-12 Education: A Scoping Review. *Review of Education*.
- Brennan, K., Balch, C., Chung, M. (2019). Creative computing 3.0. *Harvard University Press*: Cambridge. Retrieved from <http://scratched.gse.harvard.edu/guide/>
- Clements, D.H. (1995). Teaching creativity with computers. *Educational Psychology Review*, 7, 141–161.
- Clements, D.H., Gullo, D.F. (1984). Effects of computer programming on young children's cognition. *Journal of Educational Psychology*, 76(6), 1051–1058.
- Couger, J.D., Dangate, G. (1996). Measurement of creativity of I.S. products. *Creativity and Innovation Management*, 5(4), 262–272.
- Cutumisu, M., Adams, C., Lu, C. (2019). A Scoping Review of Empirical Research on Recent Computational Thinking Assessments. *J Sci Educ Technol*, 28, 651–676.
- CSTA (2016). K-12 Computer Science Framework. Retrieved from <https://k12cs.org/>
- DeSchryver, M.D., Yadav A. (2015), Creative and computational thinking in the context of new literacies: working with teachers to scaffold complex technology-mediated approaches to teaching and learning. *Journal of Technology and Teacher Education*, 23(3), 411–431.
- Dousay, T.A. (2018) Designing for Creativity in Interdisciplinary Learning Experiences. In: Persichitte K., Suparman A., Spector M. (eds) *Educational Technology to Improve Quality and Access on a Global Scale*. Educational Communications and Technology: Issues and Innovations. Springer, Cham

- Engelman, S., Magerko, B., McKlin, T., Miller, M., Edwards, D., Freeman, J. (2017). Creativity in Authentic STEAM Education with EarSketch. In *Proc. of the 2017 Technical Symposium on Computer Science Education*, 183–188. ACM.
- Ferreira, M.N.F., Gresse von Wangenheim, C., Missfeldt Filho, R. da Cruz Pinheiro, F., Hauck, J. C. R. (2019). Learning user interface design and the development of mobile applications in middle school. *ACM Interactions*, 26(4).
- Gal, L., Hershkovitz, A., Morán, A.E., Guenaga, M., Garaizar, P. (2017). Suggesting a log-based creativity measurement for online programming learning environment. In *Proc. of the 4th Conference on Learning at Scale*, 273–277. ACM.
- Glass, R.L. (1995). *Software creativity*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA.
- Gresse von Wangenheim, C., Hauck, J.C.R., Demetrio, M.F., Pelle, R., Alves, N. da C., Barbosa, H., Azevedo, L.F. (2018). CodeMaster – Automatic Assessment and Grading of App Inventor and Snap! Programs. *Informatics in Education*, 17(1), 117–150.
- Grover, S., Pea, R. (2013). Computational Thinking in K–12: A review of the state of the field. *Educational Researcher*, 42(1), 38–43.
- Grover, S., Basu, S., Schank, P. (2018). What we can learn about student learning from open-ended programming projects in middle school computer science. In: *Proc. of the 49th Technical Symposium on Computer Science Education*. ACM, 999–1004.
- Gu, M., Tong, X. (2004). Towards Hypotheses on Creativity in Software Development. In: Bomarius F., Iida H. (eds). Product Focused Software Process Improvement. *Lecture Notes in Computer Science*, 47–61, 3009. Springer, Berlin, Heidelberg.
- Haberland, G.S., Dacin, P.A. (1992). The Development of a Measure to Assess Viewers' Judgments of the Creativity of an Advertisement: a Preliminary Study. In: John F. Sherry, Jr. and Brian Sternthal (eds.). *NA – Advances in Consumer Research*, 19, 817–825.
- Henriksen, D., Mishra, P. Mehta, R. (2015). Novel, effective, whole: Toward a NEW framework for evaluations of creative products. *Journal of Technology and Teacher Education*, 23(3), 455–478.
- Hershkovitz, A., Sitman, R., Israel-Fishelson, R., Eguíluz, A., Garaizar, P., Guenaga, M. (2019). Creativity in the acquisition of computational thinking. *Interactive Learning Environments*, 27(5–6), 628–644.
- ISTE. (2020). *International Society for Technology in Education Standards*. Retrieved from www.iste.org
- Jackson, P.W., Messick, S. (1964). The Person, The Product, and the Response: Conceptual Problems in the Assessment of Creativity. *ETS Research Bulletin Series*, 2, i–27.
- Kaufman, J.C., Beghetto, R.A. (2009). Beyond big and little: The four C model of creativity. *Review of General Psychology*, 13(1), 1–12.
- Kaufman, J.C., Beghetto, R.A., Dilley, A. (2016). Understanding creativity in the schools. In: A.A. Lipnevich, F. Preckel, R.D. Roberts (Eds.), *Psychosocial Skills and School System*. Springer International Publishing, 133–153.
- Koh, K.H., Bennett, V., Repenning, A. (2011). Computing indicators of creativity. In *Proc. of the 8th Conference on Creativity and Cognition*, 357–358. ACM.
- Long, H. (2014). More than appropriateness and novelty: Judges' criteria of assessing creative products in science tasks. *Thinking Skills and Creativity*, 13, 183–194.
- 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.
- Manske, S., Hoppe, H.U. (2014). Automated indicators to assess the creativity of solutions to programming exercises. In *Proc. of the 14th Int. Conference on Advanced Learning Technologies*, 497–501. IEEE.
- Miller, S.R., Bailey, B.P., Kirlik, A. (2014). Exploring the Utility of Bayesian Truth Serum for Assessing Design Knowledge. *Human-Computer Interaction*, 29, 5–6.
- Mishra, P., Henriksen, D. (2013). A NEW Approach to Defining and Measuring Creativity: Rethinking Technology & Creativity in the 21st Century. *TECHTRENDS TECH TRENDS*, 57, 10–13.
- Mishra, P., Yadav, A. (2013). Of art and algorithms: Rethinking technology and creativity in the 21st century. *TechTrends*, 57(3), 10–14.
- Mishra, P., Henriksen, D. (2017). *Creativity, Technology & Education: Exploring their Convergence*. Springer International Publishing.
- Moreno-León, J., Robles, G. (2015). Dr. Scratch: a web tool to automatically evaluate Scratch projects. In: *Proc. of the 10th Workshop in Primary and Secondary Computing Education*. ACM, 132–133.
- Mustafaraj, E., Turbak, F., Svanberg, M. (2017). Identifying Original Projects in App Inventor. In *Proc. of the 30th Int. Florida Artificial Intelligence Research Society Conference*, 567–572.
- Myszkowski, N., Storme, M. (2019). Judge response theory? A call to upgrade our psychometrical account of creativity judgments. *Psychology of Aesthetics, Creativity, and the Arts*, 13(2), 167–175.

- O'Quin, K. (1987). Creative product scale: applications in product improvement. In *Creativity and Innovation: towards a European Network. Report of the First European Conference on Creativity and Innovation, 'Network in Action'*, organized by the Netherlands Organization for Applied Scientific Research TNO Delft, The Netherlands.
- O'Quin, K., Besemer, S.P. (1989). The development, reliability, and validity of the revised creative product semantic scale. *Creativity Research Journal*, 2(4), 267–278.
- P21. (2020). 21st century skills. Retrieved from www.p21.org
- Perez-Poch, A., Olmedo, N., Sanchez, F., Salan, N., Lopez, D. (2016). On the influence of creativity in basic programming learning at a first-year Engineering course. *International Journal of Engineering Education*, 32(5B), 2302–2309.
- Petersen, K., Vakkalanka, S., Kuzniarz, L. (2015). Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and Software Technology*, 64, 1–18.
- Ranjan, B.S.C., Siddharth, L., Chakabarti, A. (2018). A systematic approach to assessing novelty, requirement satisfaction, and creativity. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 32(4), 390–414.
- Renzulli, J.S., Reis, S.M. (1991). The assessment of creative products in programs for gifted and talented students. *Gifted Child Quarterly*, 35, 128–134.
- Rhodes, M. (1961). An Analysis of Creativity. *The Phi Delta Kappan*, 42(7), 305–310.
- Ritchie, G. (2001). Assessing Creativity. In G.A. Wiggins (Ed.), *Proceedings of the AISB symposium on AI and creativity in arts and science*, 3–11. York: The Society for the Study of Artificial Intelligence and Simulation of Behaviour.
- Robertson, J. (2005). Requirements analysts must also be inventors. *IEEE Software*, 22(1), 48–50.
- Rode, J.A., Weibert, A., Marshall, A., Aal, K., von Rekowski, T., El Mimouni, H. Booker, J. (2015). From computational thinking to computational making. In *Proc. of the Int. Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 239–250.
- Romero, M., Lepage, A., Lille, B. (2017). Computational thinking development through creative programming in higher education. *International Journal of Educational Technology in Higher Education*, 14(1), 42.
- Runco, M.A. (2003). Idea evaluation, divergent thinking, and creativity. In M. A. Runco (Ed.), *Perspectives on creativity research. Critical creative processes*. Hampton Press, 69–94.
- Runco, M.A., Albert, R.S. (2010). Creativity research: A historical view. In J. C. Kaufman & R. J. Sternberg (eds.), *The Cambridge handbook of creativity*. Cambridge University Press, 3–19.
- Runco, M.A., Jaeger, G.J. (2012). The standard definition of creativity. *Creativity Research Journal*, 24(1), 92–96.
- Said-Metwaly, S., Kyndt, E., Van den Noortgate, W. (2017). Approaches to Measuring Creativity: A Systematic Literature Review. *Creativity – Theories – Research – Applications*, 4(2).
- Scherer, R., Siddiq, F., Viveros, B.S. (2019). The Cognitive Benefits of Learning Computer Programming: A Meta-Analysis of Transfer Effects. *Journal of Educational Psychology*, 11(5), 764–792.
- Shell, D.F., Hazley, M.P., Soh.-K., Dee Miller, L., Chiriacescu, V., Ingraham, E. (2014). Improving learning of computational thinking using computational creativity exercises in a college CSI computer science course for engineers. In: *Proc. of the Frontiers in Education Conference*, 1–7. IEEE.
- Shute, V.J., Sun, C., Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review*, 22.
- Skager, R.W., Schultz, C.B., & Klein, S.P. (1966). Points of view about preference as tools in the analysis of creative products. *Perceptual and Motor Skills*, 22, 83–94.
- Snyder, H.T., Hammond, J.A., Grohman, M.G., Katz-Buonincontro, J. (2019). Creativity measurement in undergraduate students from 1984–2013: A systematic review. *Psychology of Aesthetics, Creativity, and the Arts*, 13(2), 133–143.
- Sternberg, R.J. (2015). Teaching for creativity: The sounds of silence. *Psychology of Aesthetics, Creativity, and the Arts*, 9(2), 115–117.
- Tissenbaum, M., Sheldon, J., Abelson, H. (2019). From Computational Thinking to Computational Action. *Comm. of the ACM*, 62(3), 34–36.
- Turbak, F., Mustafaraj, E., Svanberg, M., Dawson, M. (2017). Work in progress: Identifying and analyzing original projects in an open-ended blocks programming environment. In *Proc. of the 23rd Int. Conference on Distributed Multimedia Systems, Visual Languages and Sentient Systems*, 115–117.
- Voogt, J., Roblin, N.P. (2012). A comparative analysis of international frameworks for 21st century competences: Implications for national curriculum policies. *Journal of Curriculum Studies*, 44(3), 299–321.
- Walia, C. (2019). A Dynamic Definition of Creativity. *Creativity Research Journal*, 31(3), 237–247.
- Wang, K., Nickerson, J.V. (2017). A literature review on individual creativity support systems. *Computers in Human Behavior*, 74, 139–151.

- Ward, W.C., Warren, P.A. (1971). A field study of nonverbal creativity. *ETS Research Bulletin Series*.
- Westberg, K.L. (1996). The Effects of Teaching Students How to Invent. *The Journal of Creative Behavior*, 30, 249–267.
- Wing, J. (2006). Computational thinking. *Comm. of the ACM*, 49(3), 33–36.
- Yadav, A., Cooper, S. (2017). Fostering Creativity Through Computing. *Comm. of the ACM*, 60(2), 31–33.
- Zhong, B., Wang, Q., Chen, J., Li, Y. (2015). An Exploration of Three-Dimensional Integrated Assessment for Computational Thinking. *Journal of Educational Computing Research*, 53(4), 562–590.

N. da Cruz Alves is a Ph.D. student at the Graduate Program in Computer Science (PPGCC) at the Federal University of Santa Catarina (UFSC), Florianópolis, Brazil, and a research student at the initiative Computing at Schools/INCoD/INE/UFSC. She received her BSc (2017) and MSc (2019) in Computer Science from the Federal University of Santa Catarina. Her main research interests are in the area of computing education and assessment.

C. Gresse von Wangenheim is a professor at the Department of Informatics and Statistics (INE) of the Federal University of Santa Catarina (UFSC), Florianópolis, Brazil, where she coordinates the Software Quality Group (GQS) focusing on scientific research, development and transfer of software engineering models, methods and tools and software engineering education in order to support the improvement of software quality and productivity. She also coordinates the initiative Computing at Schools, which aims at bringing computing education to schools in Brazil. She received the Dipl.-Inform. and Dr. rer. nat. degrees in Computer Science from the Technical University of Kaiserslautern (Germany), and the Dr. Eng. degree in Production Engineering from the Federal University of Santa Catarina. She is also PMP – Project Management Professional and MPS. BR Assessor and Implementor.

L.H. Martins-Pacheco is a professor at the Department of Informatics and Statistics (INE) of the Federal University of Santa Catarina (UFSC), Florianópolis, Brazil. She received the degree Dr. Eng. degree in Production Engineering from the Federal University of Santa Catarina (2002). She has bachelor's degrees in Electrical Engineering (1983) and in Psychology (2010) from the Federal University of Santa Catarina. She had some administrative positions in university as head of Department of Informatics and Statistics (INE) and vice-rector. She is interested in computers in education and computers and society.