Computational thinking in pre-university education

Francisco J. García-Peñalvo
GRIAL Research Group,
Research Institute for Educational Sciences,
University of Salamanca
Paseo de Canalejas, 169
37008 Salamanca, Spain
+34 923294500 3433
fgarcia@usal.es

Juan Cruz-Benito
GRIAL Research Group,
Research Institute for Educational Sciences,
University of Salamanca
Paseo de Canalejas, 169
37008 Salamanca, Spain
+34 923294500 3433
juancb@usal.es

ABSTRACT

Several countries have usually adopted several priorities for developing ICT competences from kindergarten to secondary education. Most of them are focused on the development of key competences and/or coding skills. Although coding may be very attractive for young students and a very good practice or experience, it could be more interesting to develop students' logical thinking skills and problem-solving skills throughout programming approaches or computational thinking. This is a very exciting challenge with lots of possibilities regarding coding, robots, mobiles devices, Arduino-based application, game-based learning and so on. Thus it is very important discuss the experiences that are being developed worldwide in specialized for a with researchers that are working on this field, such as for example European Union TACCLE 3 - Coding project. This track is devoted to identify, share and valorize best practices and experiences (including technological and methodological issues) that focused on the development of computational thinking and related skills in any level of pre-university education.

Categories and Subject Descriptors

• **Social and professional topics➝Computational thinking**

• **Social and professional topics➝K-12 education**

Keywords

Computational Thinking; Coding in schools; Computational thinking skills and curriculum; Programming robots; Controlling things; Using logic; Coding skills throughout games; TACCLE 3

1. INTRODUCTION

It has recently been in the news that EEUU president Barack Obama called upon the nation to make computer science "a basic skill, right along with the three 'Rs'". This is not new and not more important because of Obama's opinion, it is a real need for future generations of workers that should know, at least, the basic laws of a computer-based society and, without demerit to humanities or social sciences, trying to reduce the current gap with STEM (Science, Technology, Engineering and Mathematics) [6] careers.

Current society is software-driven [17]. A very common situation in countries with a high rate of unemployment is they have unfilled positions for engineers and technicians for the industry and digital services. This means that a growth in the demand of positions related to technology and scientific knowledge, particularly Engineering, but not only, is not reflected in the increase of students in such university degrees.

In the European Union more than 800,000 professionals skilled in computing/informatics by 2020 are expected, many educators, parents, economists and politicians are starting to think that students need some computing and coding skills [3].

In Spain CODDII (*Conferencia de Directores y Decanos de Ingeniería Informática*) in junction with AENUI (*Asociación de Enseñantes Universitarios en Informática*) drew up the declaration for the inclusion of subjects related to science and informatics in secondary education [7] with five main recommendations: 1) all the technological subjects must be offered by all the educational centers, public and private ones; 2) the curriculum of science and informatics should be completed by all students; 3) it is absolutely necessary and urgent

teacher training in this area; 4) complementarily these skills can be developed transversally into other non-technological subjects; and 5) schools are responsible for transmitting and guide students on the importance of fostering vocations in STEM areas.

In EEUU there are different studies that recommend the creation of a well-defined set of K-12 computer science standards based on algorithmic/computational thinking concepts [36; 37].

On the other hand, new devices [2; 28; 29], from smartphones and tablets to electronic learning toys and robots, find new audiences with increasingly young children. This causes new challenges for teachers [30-32], for example how to define developmentally appropriate activities and content for children of different ages [4].

Whereas Information Technology (IT) literacy is the capability to use today's technology in one's own field, the notion of IT fluency adds the capability to independently learn and use new technology as it evolves [19] throughout one's professional lifetime. Moreover, IT fluency also includes the active use of algorithmic thinking (including programming) to solve problems, whereas IT literacy is more limited in scope.

The most frequent approach to teaching digital literacy has been to gradually encourage the learning of programming, and the term code-literacy [23] has been coined to referrer the process of teaching children programming tasks, from the simplest and most entertaining to the most complex, this way the student's progress is centered on the difficulty of the tasks and in their motivating characteristic. This means a link between the learning with the response to a stimulus instead to the child's learning and cognitive capabilities, following the traditional behaviorist theories [39].

However, there exist an alternative in the constructionism approach, yet considered by Papert [20] in his researches based on the Logo programming language, that conveys the idea that the child actively builds knowledge through experience and the related "learn-by-doing" approach to education. Papert wanted to create "a mathematics children can love rather than inventing tricks to teach them a mathematics they hate", because Papert's leitmotifs were Thinking about thinking and the freedom to achieve one's potential [34].

TACCLE 3 – Coding European Project [10-13; 35] focusses on supporting school teachers and developing their confidence to deliver the new computing curriculum including coding and computational thinking approaches.

The term computational thinking was made popular by Jeannette M. Wing [38], with her definition "computational thinking involves solving problems, designing systems, and understanding human behavior, by drawing on the concepts fundamental to computer science". Aho [1] simplified this concept defining it as the thought processes involved in formulating problems so "their solutions can be represented as computational steps and algorithms". García-Peñalvo [14] defined computational thinking as the application of high level of abstraction and an algorithmic approach to solve any kind of problems.

This track is devoted to identify, share and valorize best practices and experiences (including technological and methodological issues) that focused on the development of computational thinking and related skills in any level of pre-university education.

2. TRACK ORGANIZATION
This section describes in further detail the twelve contributions accepted to participate in this Conference track.

2.1 A survey of resources for introducing coding into schools
Within the TACCLE 3 – Coding [10; 13] European Union Erasmus+ KA2 Programme project, García-Peñalvo et al. [15] present a survey of a set of resources that can contribute to teaching programming to younger children including the most outstanding products in order to help teachers to introduce programming in pre-university studies.

2.2 From computational thinking to coding and back
Also related to TACCLE 3 project [10; 13] Koen DePryck [9] reflects about the introduction of coding in the curriculum at an early age is considered a long term investment in bridging the skills gap between the technology demands of the labour market and the availability of people to fill them. The keys to success include moving from mere literacy to active control – not only at the level of learners but also at the level of teachers. However, given the fast development of the field, one might wonder whether acquiring specific coding skills really is the essence of introducing coding early in the curriculum. We argue that the reach of ICT –including coding skills-- is much broader than STEM alone and a background in STEM is no longer a requirement for successful coding. The complex link between coding and computational thinking is the real critical success factor. He refers to TACCLE 3 as a successful approach to the implementation and valorisation of computational thinking across the curriculum.

2.3 Smart Textile objects and conductible ink as a context for arts based teaching and learning of computational thinking at primary school
The third paper of this track by Daniela Reimann and Christiane Maday [25] discuss the topic of self-made Smart Textile objects as a learning content for primary school level, towards the development of curriculum modules for project learning in the classroom as well as teacher training. It was developed in the 'Teachers Aids on Creating Contents for Learning Environments' TACCLE3 coding project [10; 13].

2.4 Computational thinking beyond STEM: an introduction to "moral machines" and programming decision making in Ethics classroom
Antonio M. Seoane-Pardo [33] describes a learning activity on computational thinking in Ethics classroom with compulsory secondary school students (14-16 years). It is based on the assumption that computational thinking (or better "logical thinking") is applicable not only

to STEM subjects but to any other field in education, and it is particularly suited to decision making in moral dilemmas. This will be carried out through the study of so called "moral machines", using a game-based learning approach on self-driving vehicles and the need to program such cars to perform certain behaviours under extreme situations. Students will be asked to logically base their reasoning on different ethical approaches and try to develop a schema of decision making that could serve to program a machine to respond to those situations. Students will have to deal also with the uncertainty of reaching solutions that will be debatable and not universally accepted as part of de difficulty, more ethical than technical, to provide machines with the ability to take decisions where there is no such thing as a "right" *versus* "wrong" answer, and potentially both (ore more) of the possible actions will bring unwanted consequences.

2.5 Developing Computational Thinking via the Visual Programming Tool: Lego Education WeDo

The fifth paper in the track by Pinto-Llorente et al. [22] presents a study that seeks to extend the existing research on the use of visual programming tools to work and develop computational thinking. We show the primary education students' perceptions of the use of the software Lego Education WeDo in the subject of natural sciences to promote the computational thinking. Authors tried to test the following hypotheses: Students will learn to build and program 3D models with Lego Education WeDo (H1), students will think creatively to solve the problems (H2), Lego Education WeDo will help pupils to know the relationship between cause and effect (H3), and the tasks developed will allow pupils to reflect about the possibilities they have and to find the correct answer (H4). Based on the result analysis there were evidences of the effectiveness of the project to increase the participants' awareness of the computational thinking.

2.6 Does computational thinking correlate with personality? The non-cognitive side of computational thinking

Román-González et al. [27] defend that computational thinking is still a poorly defined psychological construct from a psychometric point of view, due to there is no full consensus on a formal definition of computational thinking or how to measure it; and its correlations with other psychological constructs, whether cognitive or non-cognitive, have not been completely established. Authors aim to study specifically the correlations between computational thinking and the several dimensions from the 'Big Five' model of human personality: Conscientiousness, Openness to Experience, Extraversion, Agreeableness, and Neuroticism. To do so, the Computational Thinking Test (CTt) and the Big Five Questionnaire-Children version (BFQ-C) are administered on a sample (n=99) of Spanish students from 5th to 10th grade. Results show statistically significant correlations between computational thinking and: Openness to Experience ($r = 0.41$), Extraversion ($r = 0.30$), and Conscientiousness ($r = 0.27$). These results are partially consistent with the literature about the links between cognitive and personality variables, and corroborate the existence of a non-cognitive side of computational thinking. Hence, educational interventions aimed at fostering computational thinking should take into account these non-cognitive issues in order to be comprehensive and successful.

2.7 Analyzing the attitude of students towards robots when lectured on programming by robotic or human teachers

Conde et al. [8] explore the acceptance of robots as teachers for a group of K-12 students. These students attended a programming session designed to applied computational principles in different fields. Authors have analyzed how this acceptance varies with age, and also if their opinion changes if they were exposed to a real robot teacher or not. To this end, participants (N=210) were divided in two groups, one was lectured by a teacher and the other one by a Baxter robot. They used the Negative Attitudes towards Robots Scale and the Robot Anxiety Scale questionnaires to evaluate students' perception. Statistical analysis of the answers to these questionnaires is discussed in the paper, both taking into account the whole groups and another considering three subgroups based on age. Main conclusion of the study is that the use of the robot is relevant in the perception of the students about robots, but also that age is significant in the perception of robots as potential teachers.

2.8 Phogo: A low cost, engaging and modern proposal to learn how to program

González-Sancristán et al. [16], with inspiration in LOGO [20; 21], have developed the Phogo pedagogical platform, built around a low cost robot (less than $80) capable of tracing its path with a marker pen. We also present a high-level Python-based control library that allows for transparent and easy wireless communication with the robots. The approach was tested in an informal workshop with a group of teenagers without any previous self-conscious computational knowledge. As the students were attracted by the robot and the simple approach, they were able to gain some insight about abstract programming concepts such as variables, functions, and flow control structures. The majority of the students were people with physical, cognitive or intellectual disabilities and they were able to follow, enjoy and learn as any other student making this an accessible activity to everyone.

2.9 Relationship of knowledge to learn in programming methodology and evaluation of computational thinking

Rojas López and García-Peñalvo [26] present a proposal to evaluate the computational thinking skills of new students in the Division of Technologies Information and Communication of the Universidad Tecnológica de Puebla to relate the knowledge indicated in the Programming Methodology course and provide an initial environment that accredits learning, review or learn as determined by the test in order to motivate the student who already has a knowledge and attends the required education. The main conclusion of the work is to create learning scenarios through assessing the skills mentioned initially using reagents internationally recognized.

2.10 Student2student: Arduino Project-based Learning

Martín-Ramos et al. [18] present the work that jeKnowledge (Júnior Empresa da Faculdade de Ciências e Tecnologias da Universidade de Coimbra), a student-led initiative, has done in the 'jeKnowledge academy' Summer School to actively engage Portuguese high-school students in STEM education through hands-on projects based on the low-cost Arduino platform. F2F activities are complemented with tutorials and more advanced project suggestions in a blog. The course resulted in high degrees of satisfaction for both the students in the teaching role and the younger students and even influenced the Physics Department where it took place, which included Arduino workshops in the first-year syllabus of the Physics Engineering course.

2.11 Improving computational skills and curriculum of Vocational Training students. Case Study: Technological Behaviour of pharmacy students in a Digital Marketing Course

Canaleta et al. [5] aims to focus on develop an implementation proposal and its analysis according to the profiles and motivation of the students, evaluating if this innovative proposal improves the satisfaction and the academic results of the students. It has been tested in two different groups of students in Pharmacy and Para-pharmacy technician, and Technician of assistance for people in situation of dependency, both trained in a course of Digital Marketing. As demonstrated in the results, the experience has been positive, and demonstrated how the approach can be adapted to non-technological profiles without many problems.

2.12 What is the question? Impact of question length and illustration support on the success and skip rates of pre-university mock online tests

The last paper by Redondo et al. [24] analyzes the results of over 20,000 pre-university mock online quizzes in the technical drawing subject of the Spanish University Admission Test to assess the impact of (a) the presence of accompanying graphic illustrations and (b) the length of the questions on the questions success and skip rates. The performance difference of the students in two tests 6 years apart and participating in two different intakes is also discussed. The results show that questions with image support receive more successful answers and are skipped less often, while increasing question length have the opposite effect. These findings should help improving the design of online quizzes, including more visual content and/or rephrasing the questions to be more concise.

3. ACKNOWLEDGMENTS

4. REFERENCES

[1] Aho, A.V., 2012. Computation and Computational Thinking. *Computer Journal 55*, 7 (Jul), 832-835. DOI= http://dx.doi.org/10.1093/comjnl/bxs074.

[2] Alonso De Castro, M.G., 2014. Educational projects based on mobile learning. *Education in the Knowledge Society 15*, 1, 10-19.

[3] Balanskat, A. and Engelhardt, K., 2015. *Computing our future. Computer programming and coding Priorities, school curricula and initiatives across Europe.* European Schoolnet.

[4] Bers, M.U., Flannery, L., Kazakoff, E.R., and Sullivan, A., 2014. Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. *Computers and Education 72*, 145-157. DOI= http://dx.doi.org/10.1016/j.compedu.2013.10.020.

[5] Canaleta, X., Fonseca, D., Navarro, I., Climent, A., and Vicent, L., 2016. Improving computational skills and curriculum of Vocational Training students. Case Study: Technological Behaviour of pharmacy students in a Digital Marketing Course. In *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'16) (Salamanca, Spain, November 2-4, 2016)*, F.J. García-Peñalvo Ed. ACM, New York, NY, USA.

[6] Cedefop, 2015. *EU Skills Panorama (2014) STEM skills Analytical Highlight.* European Commission.

[7] Coddii and Aenui, 2014. Por la inclusión de asignaturas específicas de ciencia y tecnología informática en los estudios básicos de la enseñanza secundaria y bachillerato. *ReVisión 7*, 2, 5-7.

[8] Conde, M.Á., Fernández-Llamas, C., Rodríguez-Lera, F.J., Rodríguez-Sedano, F.J., Matellán-Olivera, V., and García-Peñalvo, F.J., 2016. Analyzing the attitude of students towards robots when lectured on programming by robotic or human teachers. In *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'16) (Salamanca, Spain, November 2-4, 2016)*, F.J. García-Peñalvo Ed. ACM, New York, NY, USA.

[9] Depryck, K., 2016. From computational thinking to coding and back. In *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'16) (Salamanca, Spain, November 2-4, 2016)*, F.J. García-Peñalvo Ed. ACM, New York, NY, USA.

[10] García-Peñalvo, F.J., 2016. A brief introduction to TACCLE 3 – Coding European Project. In *2016 International Symposium on Computers in Education (SIIE)*, F.J. García-Peñalvo and J.A. Mendes Eds. IEEE, USA.

[11] García-Peñalvo, F.J., 2016. Presentación del Proyecto TACCLE3 Coding. In *Proceedings of the Workshop EI<18. Educación en Informática sub 18* (Salamanca, España2016).

[12] García-Peñalvo, F.J., 2016. Presentation of the TACCLE3 Coding European Project. http://repositorio.grial.eu/handle/grial/654.

[13] García-Peñalvo, F.J., 2016. Proyecto TACCLE3 – Coding. In *XVIII Simposio Internacional de Informática Educativa, SIIE 2016*, F.J. García-Peñalvo and J.A. Mendes Eds. Ediciones Universidad de Salamanca, Salamanca, España, 187-189.

[14] García-Peñalvo, F.J., 2016. What Computational Thinking Is. *Journal of Information Technology Research 9*, 3, v-viii.

[15] García-Peñalvo, F.J., Rees, A.M., Hughes, J., Jormanainen, I., Toivonen, T., and Vermeersch, J., 2016. A survey of resources for introducing coding into schools. In *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'16) (Salamanca, Spain, November 2-4, 2016)*, F.J. García-Peñalvo Ed. ACM, New York, NY, USA.

García-Peñalvo, F. J., & Cruz-Benito, J. (2016). Computational thinking in pre-university education. In F. J. García-Peñalvo (Ed.), *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'16) (Salamanca, Spain, November 2-4, 2016)* (pp. 13-17). New York, NY, USA: ACM. doi:10.1145/3012430.3012490

[16] González-Sancristán, C., García-Saura, C., and Molins-Ruano, P., 2016. Phogo: A low cost, engaging and modern proposal to learn how to program. In *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'16) (Salamanca, Spain, November 2-4, 2016)*, F.J. García-Peñalvo Ed. ACM, New York, NY, USA.

[17] Manovich, L., 2013. *Software Takes Command*. Bloomsbury, New York, USA.

[18] Martín-Ramos, P., Lopes, M.J., Lima Da Silva, M.M., and Ramos Silva, M., 2016. Student2student: Arduino Project-based Learning. In *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'16) (Salamanca, Spain, November 2-4, 2016)*, F.J. García-Peñalvo Ed. ACM, New York, NY, USA.

[19] National Research Council Committee on Information Technology Literacy, 1999. *Being Fluent with Information Technology*. National Academy Press, Washington, DC.

[20] Papert, S., 1980. *Mindstorms: Children, computers, and powerful ideas*. Basic Books, New York, NY, USA.

[21] Papert, S. and Solomon, C., 1971. *Twenty things to do with a computer*. MIT.

[22] Pinto-Llorente, A.M., Casillas-Martín, S., Cabezas-Martín, M., and García-Peñalvo, F.J., 2016. Developing Computational Thinking via the Visual Programming Tool: Lego Education WeDo. In *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'16) (Salamanca, Spain, November 2-4, 2016)*, F.J. García-Peñalvo Ed. ACM, New York, NY, USA.

[23] Prensky, M., 2008. Title. In Edutopia, http://www.edutopia.org/literacy-computer-programming.

[24] Redondo, E., Regot, J., Fonseca, D., Valls, F., and Giménez, L., 2016. What is the question? Impact of question length and illustration support on the success and skip rates of pre-university mock online tests. In *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'16) (Salamanca, Spain, November 2-4, 2016)*, F.J. García-Peñalvo Ed. ACM, New York, NY, USA.

[25] Reimann, D. and Maday, C., 2016. Smart Textile objects and conductible ink as a context for arts based teaching and learning of computational thinking at primary school. In *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'16) (Salamanca, Spain, November 2-4, 2016)*, F.J. García-Peñalvo Ed. ACM, New York, NY, USA.

[26] Rojas López, A. and García-Peñalvo, F.J., 2016. Relationship of knowledge to learn in programming methodology and evaluation of computational thinking. In *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'16) (Salamanca, Spain, November 2-4, 2016)*, F.J. García-Peñalvo Ed. ACM, New York, NY, USA.

[27] Román-González, M., Pérez-González, J.-C., Moreno-León, J., and Robles, G., 2016. Does computational thinking correlate with personality? The non-cognitive side of computational thinking. In *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'16) (Salamanca, Spain, November 2-4, 2016)*, F.J. García-Peñalvo Ed. ACM, New York, NY, USA.

[28] Sánchez Prieto, J.C., Olmos Miguеláñez, S., and García-Peñalvo, F.J., 2013. Mobile Learning: Tendencies and Lines of Research. In *Proceedings of the First International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'13) (Salamanca, Spain, November 14-15, 2013)*, F.J. García-Peñalvo Ed. ACM, New York, NY, USA, 473-480. DOI= http://dx.doi.org/http://dx.doi.org/10.1145/2536536.2536609.

[29] Sánchez Prieto, J.C., Olmos Miguеláñez, S., and García-Peñalvo, F.J., 2014. Understanding mobile learning: devices, pedagogical implications and research lines. *Education in the Knowledge Society 15*, 1, 20-42.

[30] Sánchez Prieto, J.C., Olmos-Miguеláñez, S., and García-Peñalvo, F.J., 2016. *A TAM based tool for the assessment of the acceptance of mobile technologies among teachers*. GRIAL Research Group / University of Salamanca.

[31] Sánchez Prieto, J.C., Olmos-Miguеláñez, S., and García-Peñalvo, F.J., 2016. Technologically Reluctant Teachers. A TAM Based Study On Compatibility And Resistance To Change Among Pre-Service Teachers. In *Proceedings of the ECER 2016* (Dublin, Ireland, 22-26 August, 20162016). DOI= http://dx.doi.org/https://dx.doi.org/10.6084/m9.figshare.3863205.v1

[32] Sánchez-Prieto, J.C., Olmos-Miguеláñez, S., and García-Peñalvo, F.J., 2017. MLearning and pre-service teachers: An assessment of the behavioral intention using an expanded TAM model. *Computers in Human Behavior In Press*. DOI= http://dx.doi.org/http://dx.doi.org/10.1016/j.chb.2016.09.061

[33] Seoane-Pardo, A.M., 2016. Computational thinking beyond STEM: an introduction to "moral machines" and programming decision making in Ethics classroom. In *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'16) (Salamanca, Spain, November 2-4, 2016)*, F.J. García-Peñalvo Ed. ACM, New York, NY, USA.

[34] Stager, G.S., 2016. Seymour Papert (1928-2016). Father of educational computing. *Nature 537*, 308-308.

[35] Taccle 3 Consortium, 2016. TACCLE 3: Coding Erasmus + Project website. http://www.taccle3.eu/.

[36] Tucker, A., Deek, F., Jones, J., Mccowan, D., Stephenson, C., and Verno, A., 2006. *A model curriculum for K-12 Computer Science: Final Report of the ACM K–12 Task Force Curriculum Committee*. ACM, New York, NY, USA.

[37] Wilson, C., Sudol, L.A., Stephenson, C., and Stehlik, M., 2010. *Running on Empty: The Failure to Teach K-12 Computer Science in the Digital Age*. Association for Computing Machinery (ACM), New York, NY, USA.

[38] Wing, J.M., 2006. Computational Thinking. *Communications of the ACM 49*, 3, 33-35. DOI= http://dx.doi.org/10.1145/1118178.1118215.

[39] Zapata-Ros, M., 2015. Pensamiento computacional: Una nueva alfabetización digital. *RED, Revista de Educación a distancia 46*.