

## What Computational Thinking Is

Francisco José García-Peñalvo

Computer Science Department  
Research Institute for Educational Sciences  
GRIAL research group  
University of Salamanca  
JITR Editor-in-Chief

fgarcia@usal.es

### Introduction

Current software-driven Society demands skilled professionals for ICT (Information and Communication Technologies) business sector. 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 has caused an increasing approach for introduce digital or information technology (IT) literacy from the early beginning of the individual development till the high school courses (Allan, Barr, Brylow, & Hambrusch, 2010), even in post-secondary institutions (Astrachan, Hambrusch, Peckham, & Settle, 2009), combining it with other key competences such as reading, writing and math skills.

The most frequent approach to teaching digital literacy has been to gradually encourage the learning of programming, and the term code-literacy (diSessa, 2000; Hockly, 2012; Prensky, 2008; Rushkoff, 2012; Vee, 2013) has been coined to refer to the process of teaching children programming tasks, from the simplest and most entertaining to the most complex, this way the student's progress is centred on the difficulty of the tasks and in their motivating characteristic.

Consequently, at the same time that children learn human languages, both for speaking and writing, natural languages, encompassing all matters related with the experimental sciences (physics, chemistry, biology, etc.), and humanity languages, involving social sciences and humanities, it is also necessary they learn digital languages, in which ones the competences to be success in the digital world are included, using coding as the way to solve problems and computational thinking as working paradigm (Llorens-Largo, 2015).

With the awareness of the importance of digital skills and related information technology (eSkills), there are several proposals worldwide about the need to include coding from the curriculum of non-university levels, starting since primary education (or sooner) (Balanskat & Engelhardt, 2015; Brown et al.,

2013), because of the code-literacy skills are becoming understood as a core element for STEM (Science, Technology, Engineering, & Mathematics) subjects (Weintrop et al., 2016) and imaginative programming is the most crucial element of computing because it closely aligns mathematics with computing and in this way brings mathematics to life (Felleisen & Krishnamurthi, 2009). According to this different projects have been developed in different world regions such as Tackle 3 Coding European Project (García-Peñalvo, 2016a; TACCLE 3 Consortium, 2016) that provides practical ideas that teachers can use immediately together with suggestions on how these can be adapted for introducing computing or coding in their classrooms; the 985 project in China that explicitly pointed out that “prominent computational thinking ability must be the fundamental skill for innovative talents in any discipline” (Long, Zhang, & Li, 2013), or Mobile Computing project in EEUU (Turbak, Pokress, & Sherman, 2014), which is devoted to teach the big ideas of computer science to undergraduate students using App Inventor (Pokress & Domínguez Veiga, 2013). These projects are a great example of the connection of the university with the society (García-Peñalvo, 2011, 2016b).

A code-literate person means that can read and write in programming languages (Román-González, 2014), computational thinking is referred to the underlying problem-solving cognitive process that allows it. Thus, coding is a key way to enable computational thinking (Lye & Koh, 2014) and computational thinking may be applied to various kinds of problems that do not directly involve coding tasks (Wing, 2008).

The term computational thinking was made popular by Jeannette M. Wing (2006), however this increasing interest about introducing coding, code-literacy or computational thinking in the pre-university studies, there still exist a lack of consensus on a formal definition of these terms (Barr & Stephenson, 2011; Gouws, Bradshaw, & Wentworth, 2013; Grover & Pea, 2013).

Although coding is so interesting, it is more important to emphasize in the idea of computational thinking as the application of high level of abstraction and an algorithmic approach to solve any kind of problems.

From JITR, we encourage to promote coding and special computational thinking in pre-university studies and also in other university disciplines different from Computer Science.

### **Contents of the issue**

Current JITR issue comprises five papers.

The first one, “Extending the Technology Acceptance Model and the Critical Success Factors Model to Predict the Use of Cloud Computing” (Ababneh, 2016), extends the Technology Acceptance Model (TAM) (Davis, 1989) in the context of using Cloud Computing. The proposed model synthesizes the Information System Success model and the TAM to explain and predict the users’ attitudes toward using Cloud Computing.

The paper “Virtual Learning Communities in Google Plus, implications and sustainability in MOOCs” (Martínez-Núñez, Borrás-Gene, & Fidalgo-Blanco, 2016) proposes a new model to allocate informal learning and collective intelligence in Massive Open Online Courses (MOOCs) (García-Peñalvo, 2015) using external virtual learning communities through social networks, based on Google +. The main aim of this article is to assess the virtual learning community performance and analyze the interactions and the kinds of learning that take place inside the community and over time.

The paper entitled “Investment Location Selection based on Economic Intelligence and Macbeth Decision Aid Model” (Mostafa, Khaled, Jamila, & Hanoune, 2016) presents a case study that aims to apply some sound MCDM techniques in the case of Economic Intelligence (EI) and show how the use of strategic information may help deciders to choose among geographic locations in which they could settle their investments.

Salem et al. (2016) propose a system that would evaluate answers using Natural Language Processing and lastly compared the results obtain by human expert graders and proposed system.

In the last paper Winley & Singhapong (2016) examine the importance assigned by Human Resource personnel to the personality traits of Information Technology officers.

## References

- Ababneh, H. T. (2016). Extending the Technology Acceptance Model and the Critical Success Factors Model to Predict the Use of Cloud Computing. *Journal of Information Technology Research*, 9(3).
- Allan, V., Barr, V., Brylow, D., & Hambrusch, S. (2010). Computational thinking in high school courses *Proceedings of the 41st ACM Technical Symposium on Computer Science Education, SIGCSE'10* (pp. 390-391). New York, NY, USA: ACM.
- Astrachan, O., Hambrusch, S., Peckham, J., & Settle, A. (2009). The present and future of computational thinking. *SIGCSE Bulletin Inroads*, 41(1), 549-550. doi:10.1145/1539024.1509053
- Balanskat, A., & Engelhardt, K. (2015). *Computing our future. Computer programming and coding Priorities, school curricula and initiatives across Europe*. Retrieved from Brussels, Belgium: [http://fcl.eun.org/documents/10180/14689/Computing+our+future\\_final.pdf/746e36b1-e1a6-4bf1-8105-ea27c0d2bbe0](http://fcl.eun.org/documents/10180/14689/Computing+our+future_final.pdf/746e36b1-e1a6-4bf1-8105-ea27c0d2bbe0)
- Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12: What is Involved and What is the role of the computer science education community? *ACM Inroads*, 2(1), 48-54. doi:10.1145/1929887.1929905
- Brown, N. C. C., Kölling, M., Crick, T., Peyton Jones, S., Humphreys, S., & Sentance, S. (2013). Bringing computer science back into schools: Lessons from the UK *Proceeding of the 44th ACM Technical Symposium on Computer Science Education, SIGCSE '13* (pp. 269-274). New York, USA: ACM.

- Davis, F. D. (1989). Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Quarterly*, 13(3), 319-340.
- diSessa, A. A. (2000). *Changing minds: Computers, learning, and literacy*. Cambridge: MIT Press.
- Felleisen, M., & Krishnamurthi, S. (2009). Why computer science doesn't matter. *Communications of the ACM*, 52(7), 37-40.  
doi:10.1145/1538788.1538803
- García-Peñalvo, F. J. (2011). La Universidad de la próxima década: La Universidad Digital. In C. Suárez-Guerrero & F. J. García-Peñalvo (Eds.), *Universidad y Desarrollo Social de la Web* (pp. 181-197). Washington DC, USA: Editandum.
- García-Peñalvo, F. J. (2015). Massive Open Online Courses as Data Sources for Making Decisions in Learning Processes. *Journal of Information Technology Research*, 8(4), iv-vii.
- García-Peñalvo, F. J. (2016a). *Proyecto TACCLE3 – Coding*. Paper presented at the XVIII Simposio Internacional de Informática Educativa, SIIIE 2016, Salamanca, España.
- García-Peñalvo, F. J. (2016b). The Third Mission. *Education in the Knowledge Society*, 17(1), 7-18. doi:<http://dx.doi.org/10.14201/eks2016171718>
- Gouws, L. A., Bradshaw, K., & Wentworth, P. (2013). Computational thinking in educational activities: An evaluation of the educational game light-bot *Proceedings of the 18th ACM Conference on Innovation and Technology in Computer Science Education, ITiCSE '13* (pp. 10-15). New York, NY, USA: ACM.
- Grover, S., & Pea, R. (2013). Computational thinking in K-12: A review of the state of the field. *Educational Researcher*, 42(1), 38-43.  
doi:10.3102/0013189X12463051
- Hockly, N. (2012). Digital literacies. *ELT Journal*, 66(1), 108-112.  
doi:10.1093/elt/ccr077
- Llorens-Largo, F. (2015). Dicen por ahí. . . . que la nueva alfabetización pasa por la programación. *ReVisión*, 8(2), 11-14.
- Long, X., Zhang, J., & Li, Z. (2013). The Design of Teaching Structure based on Competency Training of Computational Thinking *Proceedings of the International Conference on Education Technology and Information System (ICETIS 2013)* (pp. 421-425). Amsterdam, The Netherlands: Atlantis Press.
- 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.  
doi:10.1016/j.chb.2014.09.012
- Martínez-Núñez, M., Borrás-Gene, O., & Fidalgo-Blanco, Á. (2016). Virtual Learning Communities in Google Plus, implications and sustainability in MOOCs. *Journal of Information Technology Research*, 9(3).
- Mostafa, B., Khaled, A., Jamila, E. A., & Hanoune, M. (2016). Investment Location Selection based on Economic Intelligence and Macbeth Decision Aid Model. *Journal of Information Technology Research*, 9(3).
- Pokress, S. C., & Domínguez Veiga, J. J. (2013). MIT App Inventor. Enabling personal mobile computing *Proceedings of Programming for Mobile and Touch, PProMoTo 2013*.

- Prensky, M. (2008). Programming Is the New Literacy. Retrieved from <http://www.edutopia.org/literacy-computer-programming>
- Román-González, M. (2014). Aprender a programar 'apps' como enriquecimiento curricular en alumnado de alta capacidad. *Bordón. Revista de Pedagogía*, 66(4), 135-155. doi:<http://dx.doi.org/10.13042/Bordon.2014.66401>
- Rushkoff, D. (2012). Code Literacy: A 21st-Century Requirement. Retrieved from <http://www.edutopia.org/blog/code-literacy-21st-century-requirement-douglas-rushkoff>
- Salem, S. B., Cheniti-Belcadhi, L., Braham, R., & Delestre, N. (2016). Short and Open Answer Question Assessment System Based on Concept Maps. *Journal of Information Technology Research*, 9(3).
- TACCLE 3 Consortium. (2016). TACCLE 3: Coding Erasmus + Project website. Retrieved from <http://www.taccle3.eu/>
- Turbak, F., Pokress, S. C., & Sherman, M. (2014). Mobile computational thinking with APP inventor 2. *Journal of Computing Sciences in Colleges*, 29(6), 15-17.
- Vee, A. (2013). Understanding Computer Programming as a Literacy. *LiCS. Literacy in Composition Studies*, 1(2), 42-64.
- Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining Computational Thinking for Mathematics and Science Classrooms. *Journal of Science Education and Technology*, 25(1), 127-147. doi:10.1007/s10956-015-9581-5
- Wing, J. M. (2006). Computational Thinking. *Communications of the ACM*, 49(3), 33-35. doi:10.1145/1118178.1118215
- Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society a-Mathematical Physical and Engineering Sciences*, 366(1881), 3717-3725. doi:10.1098/rsta.2008.0118
- Winley, G. K., & Singhapong, P. (2016). Personality Traits among Information Technology Officers and the Expectations of Human Resource Personnel. *Journal of Information Technology Research*, 9(3).