Pensamiento Computacional, Currículo e Interdisciplinariedad Una entrevista a Johannes Krugel*

Computational Thinking, Curriculum and Interdisciplinarity An interview with Johannes Krugel

*Daniel Andrés Quiroz-Vallejo***

https://orcid.org/0000-0003-1228-7101

Tipo de Artículo: Entrevistas

Doi: 10.17533/udea.unipluri.20.1.11

Cómo citar este artículo:
Quiroz-Vallejo, D. A. (2020). Computational Thinking, Curriculum and Interdisciplinarity. An interview with Johannes Krugel. *Uni-Pluriversidad, 20*(1), e2020110. doi: 10.17533/udea.unipluri.20.1.11

FACULTAD DE EDUCACIÓN

Recibido: 2019-11-08 • Aprobado: 2020-07-23

---

* This interview was conducted in August 2019 at the Faculty of Education of the University of Antioquia, Medellín, Colombia, and it has been reviewed by Johannes Krugel in October 2019.

** Mathematics and Physics pre-service teacher at the University of Antioquia. Member of the research group MATHEMA-FIEM
E-mail: daniel.quirozv@udea.edu.co

In Colombia, official documents aim to integrate technology in a broad and deep way. However, it is necessary to expand in this regard in educational settings. In addition, some studies show that initial teacher training can go beyond of automation or technical aspects, to generate a stronger appropriation in the integration of technology in educational processes (Carmona-Mesa, Krugel & Villa-Ochoa, 2020).

In this context, significant initiatives have been developed to transform current practices, by promoting Computational Thinking. Nevertheless, these projects have been isolated and short-term (e.g., Muñoz et al, 2019). Due to the above, it is necessary carry out new long-term projects that offer a deep analysis about their impact. In Colombia, some initiatives have focused on the implementation of Computational Thinking through virtual courses in educational institutions (Basogain et al, 2017; Basogain et al, 2018; Echeverria, Cobos & Morales, 2019); others have focused on experiences with and without computers (Sánchez, 2016; Carmona-Mesa & Cardona, 2019); also, some others are interested in establishing a curriculum that favors achieving minimum competencies in Computational Thinking (Giraldo, 2014); finally, some studies offer a general overview in public politics related to Technology teacher training in Colombia and the potential of Computational Thinking to expand their scope (Carmona-Mesa, Krugel & Villa-Ochoa, 2020).

Dr. Johannes Krugel came to Medellin as guest of the mayor's office to conduct a series of STEM+H. theme activities and a conference about Computational Thinking, Curriculum and interdisciplinarity in the University of Antioquia.

Johannes Krugel is a Doctor in Computer Science, and a lecturer and researcher at the School of Education, Technical University of Munich (TUM) in Germany. His main research interests include computer science education and computer-based learning.

Daniel: First, thank you for being here with us, thank you for accepting our invitation. We have some questions for you, motivated by our interest in getting a deeper insight in computational thinking research. So, first, we want to ask you: What kind of problems or needs in German school education encourage research and teaching strategies based on computational thinking?

Johannes: The reason for researching the implementation of computational thinking was mainly the digitalization, which transformed the world several years ago and is going to transform it even more in the future. So, the goal is to prepare the students not only to understand digitalization, but also to be able to shape it, to use it and to create their own solutions; but at the same time, give them the possibility to know the technical limits and the risks that come with such a technology. (Yeah! So, that is the main goal).

Daniel: Thank you. Also, we wanted to ask you what are the similarities and differences between computational thinking and algorithmic thinking? We often assume that they are the same thing, but we want to know if any difference between them is there.

Johannes: Yeah, sometimes I think that they are mixed or confused. But computational thinking covers just the whole process of problem solving, like representing the problem using abstract model, then finding a symbolic representation, finding algorithms to solve the abstract problem, and interpreting the results. Algorithmic thinking is just the part that has to do with algorithms, at least that is my understanding and I think it is very common. So, that is the part that has to do with algorithmic solutions… It is about building algorithms to solve those problems.

Daniel: Ok. So, we can say that algorithmic thinking is just a little part of computational thinking.

Johannes: It is a very important part but yes, it is a part.

Daniel: What aspects do you consider essential when introducing algorithmic thinking in secondary education? What should be considered before, during and after the teaching process?

Johannes: Ok. So, maybe here, we talk about computational thinking because we cover the whole topic. I think if you want to introduce this into secondary school education, maybe the most important thing is a good teacher training. So, this could be included in the regular studies of the new teachers or prospective teachers and they could receive some training on “what is computational thinking?” and “how can you implement it in schools?”. Other solution could be to offer some qualification training method for in-service teachers, so they can also be qualified, because the new teachers are only going to be teachers in several years. Then it may be good to have qualification for in-service teachers. In Germany, now, it is considered rather important to have designated subjects in school for computer science, which is called “Informatik” (it’s just a translation), because if you have separate a subject for computer science you have enough space or time in the school to actually do something, and it’s also obvious that you really need a good training for the teachers. The other option would be to integrate computational thinking in mathematics, physics, or biology but then, the training process would be probably harder to achieve because those teachers are not really familiar with computer science.

Daniel: So, do you propose to integrate computational thinking subjects, not looking at them as separated areas of knowledge?

Johannes: Yes, in Germany that is accepted in most regions. In most regions we have stated that computer science is not just computational thinking, it is also about how to use computer programs, but there is a significant part of computational thinking also there. Additionally, I think you should keep in mind that, if you want to integrate computational thinking into schools, you should keep it realistic, so that the teacher actually can do what is planned. So, computational thinking should be adapted to education subjects. Connections to other disciplines are very important to present computational thinking through many different academic areas. It would be preferable if you could actually do this in schools, maybe we are going to talk about this later: "What problems can be there?". But this is really good for the motivation of the students because some students just don't feel really attracted by computers, but are very attracted by arts and biology, so if we can integrate those subjects the motivation it's a lot higher, and they benefit a lot more from computational thinking aspects. For the organizational aspects, introducing new contents in the school is always a bit difficult because you have to drop some other contents and somebody has to decide what to drop and… (that's hard, yeah, I would propose that you don't).

Daniel: We have another question: What kind of challenges could arise while integrating computing thinking into the educational system of countries like Colombia or South America in general? How could they be overcome?

Johannes: One problem that I see here is that there is not a computer-science-teaching degree in many universities. So, you could integrate it into the curriculum of mathematics teachers; then it would be only for those teachers and it could be problematic to integrate this into a proper school curriculum… And then, talking about technical aspects, do Medellín's schools have internet and computers?

Daniel: Most of them do not. Sometimes they do not even have a computer in their classrooms… So, it is hard.

Johannes: Ok, but do they have separate computer rooms?

Daniel: Yes.

Johannes: Ok, but that is in Germany also normally, the classes do not have computers, they have one computer for the teacher but then we have computer rooms…

Daniel: In Colombia sometimes the computer room does not even exist.

Johannes: Ok, that is a problem. I think it is more severe outside the cities, in the rural areas. What I heard is that in some schools there is not Internet connection at all. This is a challenge for implementing computer science project. Nevertheless, there are several methods that had been tried internationally for teaching computational thinking without any computers, because computational thinking isn't about programming a computer, it's about a way of thinking and problem solving, so you can actually do this without any computer. There are several proposals: one of the most popular is called "Computer science unplugged" and they have a web page with lot of material.

Daniel: What kind of resources have shown the greatest potential to develop computational thinking in students? Which ones have been the most inefficient or have presented the most important difficulties?

Johannes: One general and very important aspect is to have activity so the students in the school should have to do something with the concept. It has no use to simply teach them computational thinking, but the most important thing is that they must learn. So, there must be activity on the students' side. This could be in-classroom activity or also an activity on the computer where the students do something. But still not everything has to be done with the computer because the goal is not that all students become programmers in the end because we don't need that many programmers, and because this also poses another challenge to the students, because learning to program is a little bit more difficult. It has a lot of cognitive load to be a programmer… but still it is nice to eventually try something on the computer and see if the model works and if the solution that the students designed works.

Daniel: Have you had or known experiences that help us strengthen the process of integration of computational thinking in the training of teachers from different areas? If yes, can you talk about them?

Johannes: Yeah, so far, to my knowledge, in Germany and Bavaria computational thinking is mostly in the computer science curriculum. So, I think for teachers in other areas of education they do not systematically learn about computational thinking. There are, at least, some ideas I heard about offering some courses for teachers of other disciplines, but I'm not sure how this is working at the moment. But there is a project for primary school, leaded by my colleague Katharina Geldreich. She runs a project where they develop an education for primary school teachers, so those primary school teachers who have no knowledge at all about computer science, no computer science background, learn some basics of computational thinking and then can give classes in 4^{th} grade in primary school where pupils learn some algorithmic concepts. So, this is one project I know and is implemented in about 20 primary schools in Bavaria.

Daniel: How could computational thinking help us in developing the technological and scientific competencies expected of students today?

Johannes: I think that in general the problem-solving process is very important for every discipline: biology, physics, natural sciences. Some problem-solving skills are especially important in most cases: you first need to build an abstraction of the problem and then, in the end, interpret the results, so this is important for all disciplines.

Daniel: Could you describe the way you design a course so that anyone can learn programming? What factors motivated the multipurpose courses on computational thinking? What course activities help students to relate computational thinking to other subjects of their curriculum?

Johannes: There, we maybe must go question by question. We have to distinguish a little bit, because the general course curriculum was developed by Professor Peter Hubwieser from the Technical University of Munich and this is a curriculum that he designed for secondary education in Bavaria. We use basically the same curriculum to design University lectures and fully online courses… but we maybe go to talk about this online course later. So, the idea of this curriculum is that, before you start to learn programming, it will be a lot easier if you already know several other concepts. So, the idea is that students first learn what is an object and how do algorithms in general. All this without the real programming language. Only after they learn all these things, which are mainly computational thinking competencies, they can learn to program. And this is distributed in the Bavarian school curriculum in grades from 6th to 10th. And then, we develop our online course "LOOP – Learning Object-Oriented Programming" based in the very same ideas, and this is a course which is freely available on the Internet for everybody and, there, everybody can learn programming. The course is not a pure programming course, but students are expected to learn to think how to program. So, they don't learn how to write a very small program but they learn to think, so they can write even larger programs once they finish the course, so this try to give them the fundamentals.

Daniel: You said that the course is available for everybody who wants, is this available in English or just in German?

Johannes: Yeah, for everybody who speaks German. The idea for developing the online course was that in some regions of Germany there is no computer science education at all, so the idea was to bridge that gap… because in Bavaria there is a lot of computer science education and students can take it before they begin their studies and bridge the gap.

Daniel: What course activities help students to relate computational thinking to other subjects of their curriculum?

Johannes: In the curriculum for the school in grades from 6th to 10th there are lots of possibilities to create links to other subjects. For example, in grade 6th there is a joint project with biology, in grade 7th there is often a project with physics. So, the curriculum is also quite open to integrate other kind of interdisciplinary projects.

Daniel: Last question. What trends and research lines have become established in the international literature and what are the most important future-research topics?

Johannes: One popular research topic in the past was "when can you actually start teaching computational thinking?" at which age can you start? maybe in primary school? and "what are the students or pupils able to learn at which age?". So, I think that is a rather important research topic in the moment. In general, it is a bit hard to say, because the definition of computational thinking is a bit vague. For research projects, I would always recommend to maybe focus on one aspect like abstraction or algorithmic thinking, and to work on this. What I think would also be interesting is maybe some research on how to create computational thinking interdisciplinary projects and to investigate this scientifically, because I think this gives students a great possibility to learn computational thinking.

Daniel: Again, thank you so much for being with us, thank you so much for giving this interview.

Johannes: Thank you for the invitation.

Final considerations

Although the conversation with Dr. Krugel shows that the initiatives referred to Computational Thinking in Colombia converge with aspects discussed at the international level, it also poses specific challenges in which research should be expanded both in school curricula and in teacher training. First, the potential and relevance in today's society of integrating Computational Thinking within the framework of the fourth industrial revolution is emphasized. However, it is important to link it to interdisciplinary projects that favor placing acquired skills and knowledge in function of problems that cut across the curriculum. The work by Carmona-Mesa, Krugel & Villa-Ochoa (2020) and Echeverría, Cobos & Morales (2019) are research that point to the above and are possible examples to guide future studies.

Secondly, although there are initiatives with (Basogain et al., 2018; Sánchez, 2016) or without a computer (Carmona-Mesa & Cardona, 2019), there is greater convergence in computer-related research. The above proposes important challenges in relation to the limitations in infrastructure of educational institutions in basic, secondary, and higher education in Colombia They must be resolved in a convincing manner in order to guarantee successful experiences in the integration of Computational Thinking. Infrastructure is not limited to computers; it also refers to specialized software and other peripherals that make it possible to materialize all possible work fronts linked to Computational Thinking (e.g., video game design and high-precision robot programming).

Thirdly, in addition to addressing infrastructure limitations, this interview reaffirms the prevailing need to strengthen training processes in Computational Thinking for current teachers and even more so for future teachers (Muñoz et al, 2019). The study by Carmona-Mesa, Krugel & Villa-Ochoa (2020) reports that, although there are various public policies and initiatives aimed at technology training for teachers in Colombia, there is still a great challenge to transcend training in functional and communication aspects of technology; these same authors present empirical evidence of the potential and scope of integrating Computational Thinking into initial teacher training.

Finally, it should be noted that the conference "Computational Thinking, Curriculum and Interdisciplinarity" was a space to promote discussions within the Education Faculty in the University of Antioquia about Computational Thinking and to contrast with significant experiences materialized in schools in Germany (to learn more about it, see Krugel & Hubwieser, 2017, 2018). Therefore, it constitutes a valuable resource for the readers of this interview to expand on other ideas.

To expand about it, go to:

https://www.youtube.com/watch?v=ND5ncLuqwwU – **Spanish version**

https://www.youtube.com/watch?v=dCcNyfXrDII - **English version**

Special thanks to Jaime Andrés Carmona-Mesa and Monica Eliana Cardona Zapata for their collaboration on this work. Also, to Dr. Johannes Krugel for accept the invitation and his valuables contributions.

References

Basogain, X., Olabe, J., Rico, M., Rodríguez, L., & Miguel, A. (2017). *Pensamiento computacional en las escuelas de Colombia: colaboración internacional de innovación en la educación.* Retrieved from http://recursos.portaleducoas.org/publicaciones/pensamiento-computacional-en-las-escuelas-de-colombia-colaboraci-n-internacional-de

Basogain, X., Olabe, M. Á., Olabe, J. C., & Rico, M. J. (2018). Computational Thinking in pre-university Blended Learning classrooms. *Computers in Human Behavior*, 80, 412–419. doi: 10.1016/j.chb.2017.04.058

Carmona-Mesa, J. A., & Cardona, M. (2019). Formación en el Pensamiento Computacional a través de juegos de mesa. In *XV Conferencia Interamericana de Educación Matemática* (pp. 1–8). Medellín, Colombia.

Carmona-Mesa, J. A., Krugel, J., & Villa-Ochoa, J. A. (2020). La formación de futuros profesores en tecnología. Aportes al debate actual sobre los Programas de Licenciatura en Colombia. En A. Richit & H. Oliveira (Eds.), *Formação de professores em contextos permeados pelas tecnologias digitais*. Livraria da Física: Brazil, en prensa.

Echeverría, L., Cobos, R., & Morales, M. (2019). Improving the Students Computational Thinking Skills With Collaborative Learning Techniques. *IEEE Revista Iberoamericana de Tecnologias del Aprendizaje*, *14*(4), 196-206.

Giraldo, L. (2014). *Competencias mínimas en Pensamiento Computacional que debe tener un estudiante aspirante a la media técnica para mejorar su desempeño en la media técnica de las Instituciones Educativas de La Alianza Futuro Digital Medellín* (Tesis de Maestría). Universidad EAFIT, Medellín, Colombia.

Krugel, J. & Hubwieser, P. (2017): Computational Thinking as Springboard for Learning Object-Oriented Programming in an Interactive MOOC. In: Proceedings of the IEEE Global Engineering Education Conference (EDUCON'17), Athens, Greece, 1709–1712. IEEE Press. https://doi.org/10.1109/EDUCON.2017.7943079

Krugel, J., & Hubwieser, P. (2018). Strictly Objects First: A Multipurpose Course on Computational Thinking. In *Computational Thinking in the STEM Disciplines* (pp. 73–98). Cham: Springer International Publishing. doi: 10.1007/978-3-319-93566-9_5

Muñoz, A. M., Muñoz, M. A., Huertas, L. C. A., Herrera, E., Toledo, J. J., & Ramos, D. X. (2019, October). Developing a Teacher Training Curriculum Including Computational Thinking Skills: Early Advances on a Study Focused on Colombia. In 2019 *XIV Latin American Conference on Learning Technologies (LACLO)* (pp. 8-11). IEEE.

Sánchez, L. (2016). *Comprendiendo el pensamiento computacional: experiencias de programación a través de scratch en colegios públicos de Bogotá* (Tesis de Maestría). Universidad Nacional de Colombia, Bogotá, Colombia.