

Importance of computational thinking in the training of engineers based on theories and learning models

Importancia del pensamiento computacional en la formación de ingenieros a partir de teorías y modelos de aprendizaje

Omar Iván Trejos Buriticá, PhD¹

Abstract

The target of this research was to propose, implement and bring to the classroom a model that would facilitate the learning of imperative programming in Systems Engineering and Computing based on the theory of meaningful learning (Dr. David Paul Ausubel), Discovery learning (Dr. Jemore Seymour Bruner) and the 4Q model of thinking preferences (Dr. William Herrmann). From the perspective of these three models, it was sought that, in this learning process, the meaning of programming would be simplified,

¹ Researcher, Computer and systems engineering, Universidad Tecnológica de Pereira, omartrejos@utp.edu.co

spaces would be opened for the student to discover solutions based on their own logic, students would be outlined and they would be aware of their profile and, in general, computational thinking will be assimilated as a basis for decision making. The adopted method was quantitative with written evaluations and valued in the range of 1 to 5 and qualitative from observation, dialogue and interaction between students and the research teacher. The results show that, both quantitatively and qualitatively, the proposed model is favorable to the learning process of imperative programming. It is concluded that in a learning process of these characteristics, it is worthwhile for the teacher to document himself about learning theories so that his work is rewarded with what the student learns regardless of the assessment that exists in between.

Keywords. Computational thinking, Computer programming, Learning models, Learning theories, Systems Engineering

Resumen

El propósito de la presente investigación era el de plantear, implementar y llevar al aula un modelo que facilitara el aprendizaje de la programación imperativa en Ingeniería de Sistemas y Computación basado en la teoría del aprendizaje significativo (Dr. David Paul Ausubel), teoría del aprendizaje por descubrimiento (Dr. Jemore Seymour Bruner) y el modelo 4Q de preferencias de pensamiento (Dr. William Herrmann). Desde la perspectiva de estos tres modelos se buscaba que, en dicho proceso de aprendizaje, se simplificara el significado de la programación, se abrieran espacios para que el estudiante descubriera soluciones a partir de su propia lógica, se perfilaran los estudiantes y éstos fueran conscientes de su perfil y, en general, se asimilara el pensamiento computacional como base para la toma de decisiones. El método adoptado fue cuantitativo con evaluaciones escritas y valoradas en el rango de 1 a 5 y cualitativo a partir de la observación, el diálogo y la interacción entre estudiantes y el docente investigador. Los resultados advierten que, tanto en lo cuantitativo como en lo cualitativo, el modelo planteado es favorable

al proceso de aprendizaje de la programación imperativa. Se concluye que en un proceso de aprendizaje de estas características, vale la pena que el docente se documente acerca de teorías de aprendizaje de manera que su labor se vea recompensada con lo que el estudiante aprenda allende la valoración que exista de por medio.

Palabras Clave. Ingeniería de sistemas, Modelos de aprendizaje, Pensamiento computacional, Programación de computadores, Teorías de aprendizaje.

1. Introduction.

One of the great difficulties that arise in university training is the preparation that, regarding models and learning theories, have engineering teachers who were trained as engineers but who work as teachers (Trejos Buriticá, 2012), which establishes The need for them to strengthen their purely disciplinary knowledge since engineering is the subject of their work as teachers but also to appropriate and apply learning models and theories, given that the environment where said engineering knowledge is applied is in the teaching context and Therefore, the convenience lies in the fact that all the strategies and actions adopted to improve learning will be favorable in addition to the awareness that engineering teachers have of the need to be strong on both edges (Annanth, 2016).

The tendency for engineers to form part of the teaching staff of higher education institutions, and especially systems engineers, is increasing every year (Nacional, 2016), which invites the establishment of forecasts for the engineering teacher to see his knowledge from engineering but his work from teaching and that, between one and another, he can count on the most appropriate tools to develop the specified task.

The research project that inspires this presentation seeks to make a proposal through which a methodological model can be proposed that enables the learning of programming from the adoption of actions and strategies derived from the theory of

meaningful learning, discovery learning theory and the 4Q model of thinking preferences. The presentation seeks to stage, in a systematic and organized way, the criteria that justify the specific importance of computational thinking in the training process of engineers, and particularly of systems engineers, based on the aforementioned theories and models.

The research problem of the 6-16-13 project lies in the enormous need to strengthen in engineering teachers both the disciplinary knowledge of engineering, whatever the branch of their specialization, and the theory that underlies teaching as a path through which the student transforms, updates or questions his cognitive base based on a new knowledge acquired. The novelty of this presentation is that it seeks to highlight computational thinking as an articulating element between the theories and learning models with the knowledge of systems engineering so that, together, they strengthen what the student can learn in their training cycle.

The research is justified, among other reasons, from three specific reasons: a) the high need for engineers in today's world, b) the inclination of engineers towards teaching as a job option, c) the new areas of knowledge that are strengthening different engineering programs. The presentation is based on a slightly more specific justification: a) the promotion of computational thinking as a basis for decision-making, b) the need to train critical thinking, the use of technologies and the algorithmization of solutions to problems emerging, c) the need to appropriate and apply theories and learning models from an engineering perspective.

According to the statistical information published in the Statistical Compendium of National Education of Colombia updated to 2016, if the trend of incursion of engineers in the teaching field in higher education continues, it could be expected by 2030 that half of the teachers University students may be engineers or from some area related to engineering. This makes it necessary to highlight the importance of the subject since, to the extent that it is done, a culture of deepening in engineering and training in teaching is created, so that the students are the beneficiaries.

The presentation presents some conceptual elements for reflection around the need to incorporate computational thinking, to understand and assimilate it by teaching engineers and to promote it in their classes, to relate it to different learning strategies and to articulate it with theories and models that they strengthen it, make it possible and make the path to the results established from the curricular point more expeditious. Although the 6-16-13 project has a rigorous statistical support that allows comparing the results obtained in parallel from two courses, one of study and the other of reference, this presentation raises from the theoretical the foundations to highlight what has been said previously.

Theory

In the first place, we will refer to Computational Thinking, which consists of the simplification of deliberative human logic to approach computational logic (Denning & Tedré, 2019), that is, to bring as many ways of solving a problem to the way it can be solved when modern technology serves as a means of solution based on a transformation of logic. Computational Thinking consists of three concepts that constitute it in its fundamental part:

Critical thinking. In its simplest definition, critical thinking is defined as the ability to perform analysis and evaluations of reasoning that comes from a specific topic and that is part of a defined context (Wing, 2006). Different paths enable critical thinking such as the scientific method, the accumulation of information, experience, observation and interaction with the context in question. Critical thinking always requires a situation, a set of rules, and a context in which those rules are valid. The situation may or may not be problematic, that is, it may require a solution but it may not necessarily be solvable. The set of rules can be defined by a method of representation or they can be the product of a tradition, myth or belief. The context can be defined by the variables and their behaviors and also by the way they intervene in the conditioning of the

rules. Confronting the rules with the situation and reviewing the possible relationships between them and the context from an analytical perspective is what constitutes, in its simplest essence, critical thinking.

The use of technology. The penetration of screens in today's world is such that at all times we have one of them at a very short distance. The characteristic of these times is not in the physical penetration of these screens but in the great influence that their content exerts and in the immense possibilities that are opened to the members of today's society both for the human and for the profane, to access information, to publish information, to raise concerns and to resolve them (Eady & Lockyer, 2013). Screens are the front-end of a world full of virtual options that make it necessary to adopt other positions and visions about life, human beings, the relationship between them and society in general. Being trained to live in today's world means understanding the ways in which new information and communication technologies can be used to the maximum for the advancement of knowledge, human wellbeing, information and access to data and of the future prospects of the society in which we live (Johnson & Wetmore, 2008).

The resolution and algorithmization of problems. Solving problems is one of the most necessary characteristics in these times, since in the society of the 21st century new scenarios appear with new forms of interaction, relationship and conflict (Trejos Buriticá, Imperative Programming with Language C, 2017). One way that technology provides is the algorithmization of the solutions that may arise, when it is seen that these problems are computable. It must be admitted that more and more problems are becoming computable thanks to the advancement of theories such as artificial intelligence, machine learning and data analysis. With the algorithmization of the problems, it is achieved that they are not seen from the perspective of human deliberative logic but from computational

logic to take advantage of the processing speed, the handling of large volumes of data, the large storage capacities and the high speeds in generating responses (Brown & Wilson, 2018).

For its part, meaningful learning, a theory developed and formulated by Dr. David Paul Ausubel, gives priority in learning to the meaning of knowledge and what the student already knows (Ausubel, 2010). The meaning of knowledge is the search for its meaning, that is, it answers the question, what is the use of what is learned? According to this theory, if the student (also known as an apprentice) finds a relationship between the new knowledge they receive and the context with which they relate on a daily basis, be it a theoretical, practical or experiential context, the knowledge acquires a different presence and , therefore, it begins to occupy memory spaces in the medium and long term as opposed to the instantaneity of the short term.

In this way, meaningful learning theory is based on three elements: a) prior knowledge that corresponds to the set of knowledge, specific, systematic, informal, academic, theoretical, practical or experiential, which occupies a space in the memory of the learner either in the short, medium or long term, b) the new knowledge that corresponds to that which is new, by definition, that is to say that it has recently appeared within a specific context as part of its related knowledge or that to which the learner he had not yet had the form or ability to access and that, for his brain, it is also new and c) the student's attitude that, basically, can be divided into two parts: the student's motivation to learn and the ability to establish relationships between their previous knowledge and their new knowledge (Ausubel, *The Acquisition and Retention of Knowledge*, 2012).

According to the theory of meaningful learning, the most relevant thing in a learning process is what the student already knows, which is why three reflections are reached: a) the contexts in which the learner operates influence their learning and correspond to the classroom context, institutional context and extra-institutional

or external context, the classroom context being the least influential for their learning and training, b) the student's ability to establish connections between prior knowledge and new knowledge depends on the motivation to learn and, to a large extent, that will depend (in turn) on the strategies that the teacher adopts to enable the creation of this ability in the student and c) the goal is that knowledge corresponds to a set of knowledge that are available in the medium and long term so that it can be used in situations similar to those from which it comes (which is known as skill) or in situ Actions that, being different, can be resolved with the same knowledge (which is known as competition).

Learning by discovery enables the student to "discover" the elements of the knowledge that he wants to acquire every time the teacher has adopted the strategies and actions that allow him to have a knowledge base from which he can fulfill his purpose as learning objective. It starts from the fact that knowledge is more durable and persistent when it comes from an autonomous discovery process by the learner (Bruner, 2006).

This theory was formulated by Dr. Jerome Seymour Bruner who considered that for the human being the maximum meaning of knowledge could be found in what he discovered by his own means and from the previous knowledge he already had.

In the light of this theory, knowledge is received, transformed and evaluated. It is received through the senses in a way that is carried to the brain. Subsequently, it is transformed into useful or not useful information or, also, it is left waiting to be classified (Bruner, *Acts of Meaning*, 2009). At a given moment, when it has been classified as useful information (that is, knowledge as such), it is evaluated to verify its validity when it is put into practice and applied in situations dissimilar to the nature that produces it or similar to the context of the which is derived.

The 4Q model of thinking preferences is a model formulated by Dr. William Herrmann that proposes the preference of a specific approach to see the world and interact with it (Herrmann, 2015), from four possible approaches, each one located in a different quadrant of the human being.

Quadrant A, also known as Logical, is the preferred one for those people who always want to know the reason for things, their genesis and their evolution in order to assimilate them more easily.

Quadrant B, known as Sequential, is the preferred quadrant of those people who simply comply very well with an order when it has been delivered in orderly and sequential steps so that there is no doubt in its execution. People in this quadrant do not question the provenance or root of a procedure. They simply do it step by step as indicated.

Quadrant C, social, is the quadrant of people who need to interact with others to feed back their knowledge, to make their points of view known and to find new and better ways that can solve certain problem situations. In the light of this quadrant, there is no questioning about the origin of knowledge, nor the sequentiality in the performance of actions.

Quadrant D or imaginative, is the quadrant of people who can go further, supported by their imagination, creativity and inventiveness, than other people can reach. They are idealists and often become so far removed from reality that, suddenly, they can find themselves alone looking for the most ingenious way to solve a problem, a way that they often cannot find but are convinced that it exists (Lumsdaine & Lumsdaine, 2005).

Below, Table 1 presents some observations about the contribution of these three theories to both the research and this presentation:

Tabla 1. Therorie's contributions

Theory / Model	Contribution to Proj	Contribution to presentation
	6-16-13	
Thought Computational	It enables the appropriation of computational logic from a purely practical perspective	It is the main topic of discussion and inquiry among the engineering teachers selected for the study.
Meaningful Learning	Provides tools to provide the student with criteria and adopt strategies that enable the meaning of programming, its logic and its associated thinking	It is one of the two theories that serves as a basis to be able to raise reasons that highlight the importance of logical thinking
Learning By Discovery	Provides tools for the teacher to visualize a path through which the student can be guided in the "discovery" of the knowledge required	It is one of the two theories that serves as a basis to be able to raise reasons that highlight the importance of logical thinking
Model 4Q of Preferences Thought	With its adoption, it allows students to profile themselves and, in this way, facilitates the approach of certain topics on their own terms and from their personal preferred approach.	It is the model that has been adopted to have elements of judgment that facilitate observations about the contribution of logical thinking in a programming course

Fuente: Self preparation

Materials And Methods.

For the development of the study that inspires this presentation, 30 computer programming teachers with undergraduate training in Systems Engineering and with experience of more than two years in computer programming subjects were selected. These teachers were contacted electronically through their directors and in the Valle del Cauca, Tolima and Eje Cafetero region in public universities that had engineering programs or associated or derivative programs within their academic offerings.

Contact with these teachers revolved around the questions of an instrument that, for the purpose of the investigation that is exposed, was designed. Open questions were answered that were sent by email after the first contact and that the answers were received by that same means. The instrument contained the following questions:

1. Years of experience teaching computer programming in Systems Engineering
2. Years of experience as a programmer
3. Do you know what computational thinking is?
4. Do you know what meaningful learning is?
5. Do you know what discovery learning is?
6. Do you know what the 4Q model consists of?
7. Do you use any of these models in the planning and development of your programming subjects?
8. If yes, explain how you do it and which model you use.
9. Are you a programming logic, programming or programming language teacher?
10. Do you consider it important for a programming engineer teacher to know about theories and models of learning and of thinking preferences?
11. Age and Sex

Questions 1 and 2 are completely closed since your answer refers to a specific value. From question 3 to question 6, although the answer could be a simple YES or NO, the teacher is invited to briefly expand it in the introduction of the instrument. In this way it is possible to try to detect if the teacher really knows the theory for which she is asking or not.

Question 7 is a closed question whose answer refers to a YES or a No. Question 8 opens a space where the teacher has the option to explain, as broad as he or she needs or as brief as he considers it. Question 9 raises the teacher the concern about her classification as a programming teacher, if what she has written so far is coherent and truthful, she will find it very easy to answer this question. If it is not, you will see yourself in tight tights to do it. Question 10 reveals,

in a very simple way, the relevance that an engineering teacher in programming areas gives to education science training and its derived models and theories.

The information collected was grouped and tabulated according to the responses collected. It must be accepted that this quantitative inquiry does not correspond to a detailed study but to a preliminary inquiry that, due to the randomness of the selected sample of teachers, could shed some light on the conception that engineering teachers have of programming subjects in engineering programs of public university systems and in similar or derived programs, on the relevance that these confer to computational thinking and the need to know models and learning theories with the aim that their efforts as teachers are more effective both in terms of learning as in relation to the time used for it.

Results And Discussion

The results obtained in this investigation are summarized in Table 2.

Table 2. Results

Prg	Enunciado	Respuestas	
1	Years of experience teaching computer programming in systems engineering	< 5	≥ 5
		22	8
2	Years of experience as a programmer	< 5	≥ 5
		25	3
3	Do you know what computational thinking is?	Yes	No
		2	28
4	Do you know what meaningful learning is?	Yes	No
		1	29
5	Do you know what discovery learning is?	Yes	No
		3	27
6	Do you know what the 4Q model consists of?	Yes	No
		1	29

		1	29	
	Do you use any of these models in the planning and development of your programming subjects?	Yes	No	7
8	How do you do it and what model do you use?	(See table 3)		
	Are you a teacher of programming logic, programming or programming languages?	LogPrg	Prog	LangPrg
		4	9	17
	Do you consider it important for a programming engineer teacher to know about theories and models of learning and preferences?	Yes	No	
10		8	22	
		thinking		
		Avg	M	F
11	Average Age and Sex	Age		
		26	19	11

Fuente: Self elaboration

Table 3. Model's use

Ans	Description
1	I try to find a relationship between the experiences of the students and the statements that are raised so that they can be solved with programming. I understand that this is what they call meaning. For example, students build a program that allows them to register the ticket on the bus and say how the returns should be delivered.
2	For two semesters I tried to place the students in the preferred quadrants to get to know them a little better. I suppose that to study programming a Logical preferred quadrant is required, although after this period I began to think about the benefits that sequential, social and imaginative students had as part of the work groups.

Despite the fact that the present investigation is not an exhaustive study, in any case due to the randomness of the sample and geographic distribution, the information collected allows us to raise some reflections in this regard that are worth considering.

Question 1 shows an overwhelming majority of teachers who have less than 5 years of experience as professors of programming in systems engineering and in question 2 it is observed that the

experience as programmers is also the majority, which is less than 5 years. If the average age of question 11 is taken into account, which corresponds to 26 years, it can be concluded that they are recently graduated systems engineers whose knowledge revolves around purely disciplinary knowledge but who, in the absence of experience, have not yet had enough time to assess the importance of other knowledge in the case of teaching in a field such as programming in systems engineering.

Questions 3, 4, 5 and 6 that inquire respectively about computational thinking, meaningful learning, discovery learning and the 4Q model, support what has been said in the previous paragraph since the majority answer is NO when asked if they know some of the these theories and / or models. The average of negative responses is 28.25, which, taken to a percentage level, corresponds to 94.1% and, therefore, it can be thought that only an approximate 6%, of the selected sample of teachers, could attest to your knowledge or at least a brief understanding around the theories and models mentioned. Indeed, the affirmative answer corresponds to an average of 1.75, which, as a percentage, is 5.83% of the teachers who filled out the form.

This leads us to think a) that the randomly selected teachers are mostly so young or so recently graduated that they have not yet been able to assess the need for other knowledge that complements disciplinary knowledge in teaching work, b) that the Theories and learning models are not a topic that interests programming engineering teachers, c) that they may know other theories and learning models but that, exactly, they do not know about those that are asked in the instrument, d) that they are possibly making enormous efforts to change the cognitive base of their students when trying to teach programming and that, one might think, these efforts could become more effective if they were based on the theories that were selected as well as the sense of training by competencies that requires the learning of modern programming, of the need for the student to relate disciplinary knowledge to real life as suggests the theory of meaningful learning, that the student may have tools that

allow him to discover knowledge relevant to programming to be able to practice it later, that a preferred profile of the 4Q model could be more convenient in the selection of students for a program such as systems engineering and that computational thinking can be the basis for programming, and knowledge of other areas, to acquire a much more applicative meaning in today's society.

As expected, the use of the specified models and theories is conspicuous by its absence in the academic scene of the selected teachers, since question 7 presents some results where it attests that only 2 teachers use these models in the planning and development of their programming subjects and that 28 teachers do not, which can be explained by the reasons stated in the immediately preceding paragraphs.

In question 8, Table 2 presents the way in which teachers have used theories and learning models. The only two teachers who use them, one of them has thought about the concept of meaning (as proposed by the theory of significant learning) when wanting to relate programming to the needs of the students' everyday world and the other has considered it important to outline to students with the conviction that those who have a Logical preferred quadrant can be more productive in terms of programming logic and therefore in terms of programmable solutions that can be implemented on a computer.

In both cases, there is evidence of a relationship, albeit timid, of the teachers from their own knowledge of systems engineering with two of the theories raised in this inquiry. It would be convenient, in a more exhaustive study, to know the results obtained by the teachers who answered YES and to make a comparison with the results of other courses where, the same teacher teaching the same subject, had not adopted the changes mentioned in their answers.

Question 9 embodies a trap door that consists of knowing if teachers really know the difference between being a professor of programming logic, programming or programming languages. The

majority answer is that teachers define themselves as programming language professors, which is hardly natural among recently graduated engineers who consider programming simply as the way to learn to handle programming languages, ignorance of the importance of the mathematical foundations that they teach. Underlie (as derived from programming logic) or programming paradigms from a perspective of their theoretical conception (as can be inferred from programming as such).

This indicates the great need to appropriate, assimilate, apply, feed back and evaluate computational thinking as a way to develop solutions that have a scientific foundation (logic), that can be located with good theoretical bases within a paradigm (programming) and that is also seen working on a computer (programming languages).

Question 10 has an answer that could be considered consistent with what has been said so far, because of the 30 randomly selected teachers in different regions and different public universities, only 8 consider it important for an engineering teacher to know about the learning models and theories and apply them in your subject. The vast majority, that is, 22 teachers (73%) do not see the relevance of these theories and models as a complement to facilitate, simplify and make their work more effective as teachers of programming in systems engineering.

4. Conclusions

Bearing in mind that the objective of the unfunded research project that inspired this presentation consisted in the Development of a methodological model for learning programming in systems engineering based on meaningful learning, discovery learning and the 4Q model of student preferences. Thinking, and that the objective of this presentation was to stage, in a systematic and organized way, the criteria that justify the specific importance of computational thinking in the process of training engineers, and particularly of systems engineers, taking As a basis for the aforementioned theories

and models, and taking into account the electronic interaction carried out with the students selected for the inquiry that supports this presentation, the following conclusions can be drawn:

Teaching is part of one of the possible occupational profiles of engineers since it is not far from their job expectations. Teaching engineers require that they be made aware of the importance of appropriating and applying theories and learning models so that their teaching work becomes more effective and efficient.

Computational thinking corresponds to the strongest conceptual edge that must be taken into account within the processes of the formation of the logic required to assimilate and apply computer programming and its associated subjects. Theories such as meaningful learning and discovery learning and models such as 4Q of thinking preferences, pave the way for teaching computer programming.

Carrying out a study with a larger sample may reveal the current state of perception of computer programming engineering teachers in systems engineering in relation to computational thinking, meaningful learning, discovery learning and the 4Q model. , the way they can take advantage of it for the development of their classes and the potential that these can awaken in the teacher in the teaching and learning process with the respective students

References

- Annanth, M. (2016, 23 March). Humanities and Engineering Education. Recuperado el 18 de 08 de 2018, de <http://www.t5eitm.org/2016/03/importance-humanitiesengineering-education/>

- Ausubel, D. (2010). *Sicología Educativa: Un enfoque cognitivo*. New York: McGraw Hill.
- Ausubel, D. (2012). *The Acquisition and Retention of Knowledge*. Washington - USA: Springer.
- Brown, N., & Wilson, G. (2018). Ten quick tips for teaching programming. *PLos Comput Biol*, 14(4), 1-8.
- Bruner, J. (2006). *Hacia una teoría de la instrucción*. México: Editorial Limusa.
- Bruner, J. (2009). *Actos de Significado*. Madrid: Alianza Editorial.
- Denning, P., & Tedré, M. (2019). *Computational Thinking*. Massachussets: The MIT Press.
- Eady, M., & Lockyer, L. (2013). *Tools for Learning: technology and teaching*. Wollongong (Australia): University of Wollongong Press.
- Herrmann, W. (2015). *The whole brain bussiness book*. New York: McGraw Hill.
- Johnson, D., & Wetmore, J. (2008). *Technology and Society: building our sociotechnological future*. Boston: The MIT Press.
- Lumsdaine, M., & Lumsdaine, E. (2005). *Creative Solving Problem*. Miami: McGraw Hill Companies.
- Nacional, M. D. (2016). *Compendio Estadístico de la Educación Superior Colombiana 2015*. Bogotá: Imprenta Nacional de Colombia.
- Trejos Buriticá, O. (2012). *Significado y Competencias*. Pereira: Editorial Papiro.

Trejos Buriticá, O. (2017). *Programación Imperativa con Lenguaje C*.
Bogotá: ECOE Ediciones.

Wing, J. (2006). Computational Thinking. *Communications on the
ACM*, 49(3), 33-35