

Interpretation of computational thinking evaluation results for enrollment prediction

Arturo Rojas-López
Information Technologies Division,
Technological University of Puebla
Puebla, México
arturo.rojas@utpuebla.edu.mx

Francisco J. García-Peñalvo
GRIAL Research Group,
Research Institute for Educational
Sciences, University of Salamanca
Salamanca, Spain
fgarcia@usal.es

ABSTRACT

During two generations (2016 and 2017) the computational thinking evaluation has been carried out in order to establish learning scenarios for new students, such interventions have been made in the Programming methodology course, it belonging to the career of Information Technology at the Technological University of Puebla in México. The results have led a personalized education for students, recognizing previous skills as well as trying to correct those missing, so that it acquires the competences respective, credit the course and improve the retention percentage of the first quarter. In this sense, when detecting possible skill gaps, is it possible to predict what will be the impact to maintain or decrease enrollment during and the end of quarter? The present work aims to answer the question by the results interpretation obtained from the computational thinking evaluation to 242 new students, generation 2018. Initially, it was established which would be the student's situation during and the end of four months from September to December based on the correct assessment reagents; three categories were determined: 1. Sure desertion, 2. Safe permanence, 3. Variable permanence. Later, 50 students who enrolled the next quarter (January-April 2019) were revised if they had been predicted properly; using a survey, the familiarity of key concepts of the subject Programming methodology was obtained with the aim of determining a correspondence with the evaluation of computational thinking skills, as well as the established situation, consequently, establishing the validity of predicting the enrollment.

KEYWORDS

Computational thinking, Problem solving, Higher education, Programming teaching, Programming Learning.

1 Introduction

Computational thinking is strongly linked to problem solving [1-3] since its initial use by Wing [4]. The concept definition is associated with computer science itself [5-10]. Some authors have determined certain skills that when promoted in pre-university studies benefit students who choose a STEM profession [11, 12](Science, Technology, Engineering and Mathematics) such as abstraction, decomposition, heuristic reasoning, planning, programming, recognition patterns and algorithms design [13-15]. The research that has been done for higher education, have reported results associated with improving the ability to solve problems in the first university studies, through exercises computational thinking in careers of engineering and computer science [16-20] in the early computer programming courses. Despite such actions, these courses represent the enemy to defeat to reduce school dropout in universities [21].

Román et al. [22] provide an important result to show that it is possible to detect students with computer talent in the academic high school level, they can have a better performance in "education standards Informatics between 1 and 2 years compared to regular students."

The results at the Technological University of Puebla (UTP) [23,24] to determine a learning scenario in the Programming methodology course, evaluating computational thinking for two generations (2016 and 2017), have led to a personalized education for students, recognizing previous skills as well as trying to correct those that are lacking in order to acquire competences respective, credit the course and improve the retention percentage of the first quarter. In this sense, when detecting possible skill gaps, is it possible to predict what will be the impact to maintain or decrease enrollment during and the end of the quarter? The

present work aims to answer the question through the results interpretation of the computational thinking evaluation in the students from generation 2018. The paper is organizing in the next sections. Context section include the problem and historical data. Description section contain the data organization from computational thinking evaluation with the objective of create prediction context. Results section comments observations and interpretation of prediction context. Finally, conclusions about the activity and data validation for future work with students of next generations.

2 Context

In the UTP's Information Technologies division, the percentage of dropout (PD) determined by the number of students enrolled in the second semester compare with enrollment in the first from 2009 to 2016, is indicated in Table 1, which illustrate an increasing trend (see Figure 1). In the aforementioned period, the average is 31.13%. From the analysis of the computational thinking evaluation results, the intention is to predict a percentage of desertion for the following generations, specifically for generation 2018.

Table 1: Student dropout rates from 2009 to 2016

Year	Percentage of dropout PD
2009	24.32
2010	27.81
2011	31.13
2012	31.14
2013	35.20
2014	42.24
2015	30.30
2016	34.00
Median	31.13

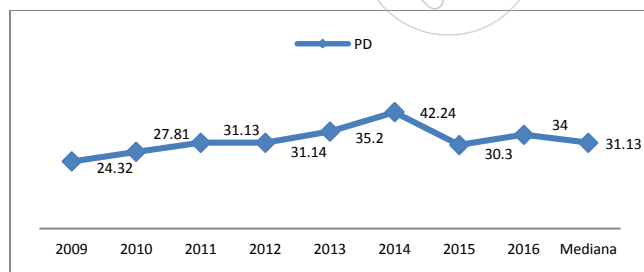


Figure 1: Student dropout percentage trend from 2009 to 2016

3 Description

The 2018 generation was 325 students enrolled. The face-to-face evaluation of computational thinking was carried out by 242 new students during the days 21 and 22 of August 2018, assistance

schedules were organized for both school shifts in the laboratories. The distribution of participating students by group is indicated in Table 2. The evaluation has 5 reagents associate with 5 skills, this means, abstraction, evaluation, decomposition, algorithm design and generalization.

Taking as reference the correct reagents of the computational thinking evaluation, 5 groups were generated.

Group 1

Students who did not obtain a correct answer from the five reagents, their profile was considered highly questionable because they apparently lacked problem solving skills.

Table 2: Number of students who made the computational thinking evaluation by group

Group	Day	Quantity
A	21	29
B		27
C		21
F		22
H		25
G	22	23
D		28
E		24
J		33
I		10
Total		242

Group 2

Students who obtained the correct answer of the five reagents, an adequate profile was considered for the study of the career because they evidencing skills in solving problems.

Group 3

Students who obtained only one correct reagent, it was considered that they could be potential low enrollment in the first months or at the end of quarter.

Group 4

Students who obtained only one incorrect of the five reagents, they were considered good candidates to study the university career.

Group 5

Students who obtained two or three correct reagents, they corresponded to young people who required academic work in two senses, to increase their computational thinking skills and to keep enrolled at the end of the quarter at least approving the Programming methodology course.

Taking as a reference the description of each determined group, three categories are predicted for enrollment at the end of the quarter with respect to the Programming methodology course.

- Those who represent a very probable school dropout. Group 1 and 3.
- Those that will be maintained for the next quarter. Students of group 2 and 4.
- Those whose permanence is variable due to the academic work that they require to maintain their

motivation or low frustration in problems resolution, difficulties or learning complexity. Group 5.

At the second quarter beginning, January – April 2019, students answered the survey indicated in Table 3. The objective was to know the concepts learned of each student from the learning units in the previous quarter, this means, to know the concepts that are familiar to students. Table contains an abbreviation of each concept.

4 Results

With the computational thinking evaluation results the following information was obtained: nine students of the 242 formed part of group 1, twelve were grouped to the second, 48 were part of group 3, 37 integrated group 4 and group 5 had the largest number of students with a total of 136 of 242. Table 4 contains the values of the information previously commented. In addition, Figure 2 shows the percentages of all students for each group and their specific descriptions, in Figure 3 the number of students expected to be enrolled is illustrated, that is, the prediction at the end of the course for the quarter September-December 2018.

Table 3: Survey for second semester students

Question	Response option
Full name	Open
Group	Selection A to J
Select the concepts which you are familiar - Multiple selection	
Data type (DT)	
Creation of identifiers for variables (ID)	
Arithmetic operators (AO)	
Logical operators (LO)	
Relational operators (RO)	
Operator hierarchy (OH)	
Solve arithmetic, logical and relational expressions (EX)	
Use of a variable accumulator and counter (CyA)	
Selection structure (SS)	
Loop structure (LS)	
Definition and creation of an algorithm (AL)	

Table 4: Distribution of students by group, description and prediction of enrollment

Group	Reagents correct	Students (242)	Description	Registration prediction
1	0	9	Highly questionable profile	Dropout - 57 - 23.55% - Category A
3	1	48	Potential dropout	
2	5	12	Appropriate profile	Safe - 49 - 20.24% - Category B
4	4	37	Good candidates	
5	2 or 3	136	Academic work	Variable - 136 - 56.20% - Category C

Only some students from three groups answered the familiarity concepts survey, giving a total of 50 from the groups A, C and D (16, 19 and 15 respectively; groups B, E to J no participating), Figure 4 shows the percentages distribution.

Of the 50 students, 11 did not perform the computational thinking evaluation. The distribution of 39 students by group and category of prediction is indicated in Table 5.

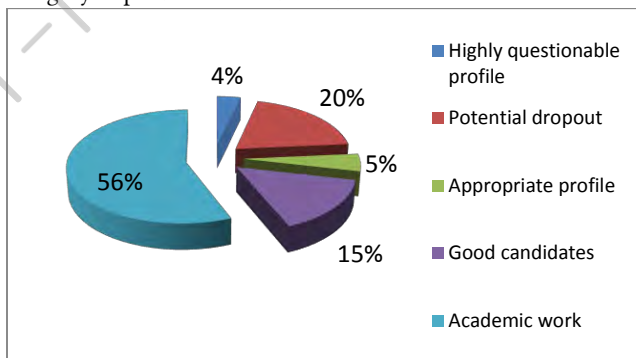


Figure 2: Distribution of students by group based on computational thinking evaluation

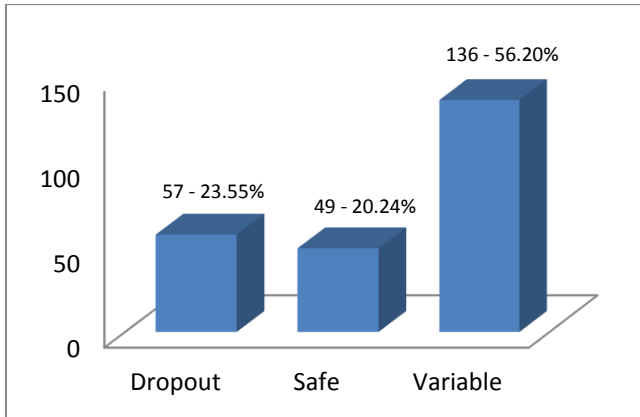


Figure 3: Enrollment prediction at the end of the semester September - December 2018

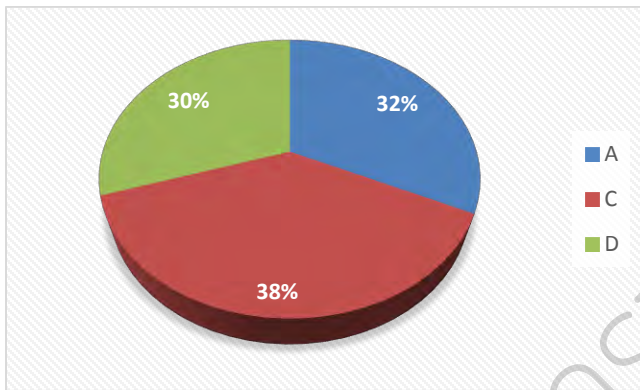


Figure 4: Percentages by groups that carried out the survey in the second quarter

From a practical point of view, it would only be advisable to revise per student the erroneous predictions of school dropout. If they answered the familiarity survey then they are enrolled in the second quarter and contradict what was expected. Unlike predictions of safe and variable enrollment that confirm and benefit the prediction. In spite of this, the results are reviewed from the survey carried out by the category and group to which it was predicted.

Category A - Desertion, groups 1 and 3.

The students that belonged to this category, initially represent an error of the prediction, were predicted as highly questionable profile and low potential; even so, they accredited the course and enrolled in the second quarter. The answers group review to the familiarity survey with the correct reagents in computational thinking evaluation had the following observations to consider.

Group 1

For the two students enrolled in the next quarter period, their study profile remains highly questionable because there is correspondence with the unfamiliarity of concepts with the background of having a lack of computational thinking skills, no more than three concepts were marked and corresponds to the

basic concepts of the course. In this sense the prediction was correct. The Table 6 shows how there is a right conclusion based in information from two students in this group.

Group 3

Eight students belonged to the group. One student indicated that he is only familiar with two concepts of the 11 and only the reagent regarding abstraction (in computational thinking evaluation) was correct, the prediction was congruent. Four students indicated to be familiar with the concepts of the second learning unit (arithmetic, logical and relational operators), but not with the evaluation of expressions; also familiar with the data types, but with respect to the competence for the design of algorithms they only indicated knowing the concept without being able to create them, they do not know the structures of selection control, repetition, counter and accumulator; what corresponds to having only obtained a correct reagent from the evaluation of computational thinking, the prediction was congruent. Three students do not express a congruence between the evaluation of computational thinking (a single correct reagent) and familiarity that they indicate, all concepts, in this sense the prediction was erroneous. The Table 7 details information.

Category B - Safe, group 2 and 4.

Only one student belonged to group 2 and corresponded his familiarity of knowledge with the five correct reagents. The prediction of safe registration is correct.

Group 4 with 7 students corresponded with the prediction, they were relevant candidates to study the career (only one erroneous reagent), despite having indicated a lack of familiarity with some concept (particularly the complex learning unit 3).

Category C - Variable, group 5

Group 5 was the easiest to verify the validity of the prediction. With academic work was considered that students could demonstrate and acquire relevant course's skills profile, 16 students got it. In addition, four students indicated familiarity in all concepts and obtained three or two correct reagents in computational thinking evaluation.

Table 8 summarizes the data of correct predictions and erroneous category and group.

Table 5: Distribution by group and category of the students surveyed

Group	Quantity	Category	Description
1	2	A	Dropout
3	8		
2	1	B	Safe
4	7		
5	21	C	Variable

Table 6. Information of concept familiarity and results computational thinking evaluation for group 1 students.

Group 1 student	Concept mark with familiarity	Right reagent - Skill
1	DT, AO, LO	None
2	DT, LO	None

Table 7. Information of concept familiarity and results computational thinking evaluation for group 3 students.

Group 3 student	Concept mark with familiarity	Right reagent - Skill
1	DT, LO	1 – Abstraction
2,3,4,5	DT, AO, LO, RO	1 – Evaluation
6,7,8	All concepts	1 – Algorithm design

Table 8: Right and wrong predictions

Category	Group	Quantity	Right	Wrong
A-Dropout	1	2	2	0
	3	8	5	3
B-Safe	2	1	1	0
	4	7	7	0
C-Variable	5	21	21	0

With the above information, 33 of the 39 students had a correct prediction, in other words, prediction had a success of 92.31% or an error of 7.69%. In this sense, with the interpretation of results of the computational thinking evaluation, a 23.55% dropout rate would have been possible in the 2018 generation with a success of 92.31%.

5 Conclusions

The activity of predicting enrollment for the generation 2018, from the computational thinking evaluation for 242 students drawn the following conclusions:

The forecast of 23.55% dropout, despite being below the average in the period from 2009 to 2016, is favorable for the University considering that it would have the guarantee of achieving a 76.44% retention at the end of the first quarter. The predicted retention percentage depended on hard academic work with the group determined as a variable (56.20%).

The actual enrollment values were 325 students in the first quarter and 202 for the second. It represents a 62.15% retention or 37.85% dropout. The determined prognosis represents a possibility of improvement to be able to have only the loss of 57 students instead of 123. It is important to emphasize that within the forecast there is a high recommendation of academic work to be able to achieve it, in the case of study it represented a population of 136 students that required an individualized follow-up.

The 92.31% of having a correct prediction is based only by 16.12% of the population that made the evaluation of computational thinking, which represents a weakness of the work done.

The future work of the research includes the application of the experiment designed for the 2019 generation, expanding the student population, especially for those who answer the familiarity survey with the aim of increasing the confidence of the data analyzed and their respective interpretation.

ACKNOWLEDGMENTS

The present work is carried out within the Doctorate program in Education in the Knowledge Society of the University of Salamanca, Spain [25-27] and the GRIAL research group [28,29], as well as the support of the Technological University of Puebla.

REFERENCES

- [1] C. Zhang, X. Chen, and L. Li, 2011. Research of VB programming teaching based on the core of computational thinking ability training, 6th International Conference on Computer Sciences & Education (ICSE), Singapore, 2011, pp. 1260-1263. DOI: <http://dx.doi.org/10.1109/ICSE.2011.6028861>.
- [2] Z. Ni, 2017. Discussion on Case Teaching Method Based on Computational Thinking in Programming Teaching. International Conference on Social Science, Education and Humanities Research (ICSEHR 2017). DOI: <http://dx.doi.org/10.2991/icsehr-17.2017.9>.
- [3] P. Compañ -Rosique, R. Satorre -Cord, F. Llorens-Largo, and R. Molina-Carmona, 2015. Teaching to program: a direct way to develop computational thinking. Journal of Distance Education, 46 (11). DOI: <http://dx.doi.org/10.6018/red/46/11>.
- [4] J.M. Wing. 2006. Computational Thinking. Communications of the ACM 49, 3, 33-35. DOI: <http://dx.doi.org/10.1145/1118178.1118215>.
- [5] M. Romero, A. Lepage, and B. Lille, 2017. Computational Thinking development through creative programming in higher education. International Journal of Educational Technology in Higher Education. 2:42. DOI: <http://dx.doi.org/10.1186/s41239-017-0080-z>.
- [6] Q. Gao, 2014. The computational thinking-oriented inquiry teaching mode for advanced programming language course. Bio Technology: An Indian Journal. Volume 10, Issue 12, pp. 6587-6595. ISSN: 0974-7435.
- [7] Z. Yinnan, and L. Chaosheng, 2012. Training for computational thinking capability on programming language teaching, 7th International Conference on Computer Science & Education (ICSE), Melbourne, VIC, 2012, pp. 1804-1809. DOI: <http://dx.doi.org/10.1109/ICSE.2012.6295420>.
- [8] L. Ying, and L. Pingping, 2017. Research on the teaching of programming language based on Computational Thinking. Proceedings of the 2017 International Conference on Social Science, Education and Humanities Research. DOI: <http://dx.doi.org/10.2991/icsehr-17.2017.17>.
- [9] C. Zhi -Mei, and L. Xiang, 2016. The PBL teaching method based on computational thinking in C programming. 2 nd International Conference on Modern Education and Social Science. ISBN: 978-1-60595-346-5, pp.405-409.
- [10] A. Xia, 2016. On the Basis of the Program Design Teaching and Research of Cultivation of Computational Thinking Ability. Proceedings of the 2nd International Conference on Social Science and Higher Education. Atlantis Press. DOI: <http://dx.doi.org/10.2991/icsshe-16.2016.83>.
- [11] E. Freudenthal, A.N. Ogrey, M.K. Roy, A. Siegel. 2010. A computational introduction to STEM studies. IEEE Education Engineering Conference, EDUCON 2010, art. no. 5492514, pp. 663-672. DOI: <http://dx.doi.org/10.1109/EDUCON.2010.5492514>.
- [12] F. J. García-Peñalvo, and J. A. Mendes. 2018. Exploring the computational thinking effects in pre-university education. Computers in Human Behavior. 80, 407-411. DOI: <http://dx.doi.org/10.1016/j.chb.2017.12.005>.
- [13] W. Huang, Z. Deng, and D. Rongsheng, 2009. Programming Courses Teaching Method for Ability Enhancement of Computational Thinking, International Association of Computer Science and Information Technology - Spring Conference, Singapore, 2009, p. 182-185. DOI: <http://dx.doi.org/10.1109/IACSIT-SC.2009.52>.
- [14] G. Chen, 2017. Programming Language Teaching Model Based on Computational Thinking and Problem-based Learning. 2nd International Seminar on Education Innovation and Economic Management (SEIEM 2017). Atlantis Press. DOI: <http://dx.doi.org/10.2991/sciem-17.2018.31>.
- [15] G. Michaelson, 2015. Teaching programming with computational and informational thinking. Journal of Pedagogic Development. Volume 5, Issue 1, March 2015. Retrieved from <https://www.beds.ac.uk/jpd/volume-5-issue-1>

- march-2015/teaching-programming-with-computational-and-informational-thinking.
- [16] J.A. Joines, D. Raubenheimer, A. Craig, 2010. Using computational tools to enhance problem solving. *Computers in Education Journal*, 20 (4), pp. 101-112.
- [17] D.B. Larkins, and W. Harvey, 2010. Introductory computational science using MATLAB and image processing. *Proceedings Computer Science*, Volume 1, Issue 1, 2010, Pages 913-919, ISSN 1877-0509, DOI: <http://dx.doi.org/10.1016/j.procs.2010.04.100>.
- [18] I.F. Swaid, 6th International Conference on Applied Human Factors and Ergonomics (AHFE 2015) and the Affiliated Conferences, AHFE 2015 Bringing Computational Thinking to STEM Education, *Procedia Manufacturing*, Volume 3, 2015, Pages 3657-3662, ISSN 2351 -9789, DOI: <http://dx.doi.org/10.1016/j.promfg.2015.07.761>.
- [19] D. Kafura, D. Tatar, 2011. Initial experience with a computational thinking course for computer science students. SIGCSE'11 - Proceedings of the 42nd ACM Technical Symposium on Computer Science Education, pp. 251-256. DOI: <http://dx.doi.org/10.1145/1953163.1953242>.
- [20] M. Roman, JC Perez, C. Jimenez C. 2015. Test of Thought Computer: General design and Computational psicometría Thinking Test: design & general psychometry, III International Congress on Learning, Innovation and Competitiveness (CINAIC 2015), October 14-16, 2015, Madrid, SPAIN.
- [21] C. McInerney. 2010. Having fun with computer programming and games: Teacher and student experiences. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5941 LNCS, p. 136-142. DOI: http://dx.doi.org/10.1007/978-3-642-11376-5_13.
- [22] M. Román-González, JC Pérez-González, J. Moreno-León, G. Robles. 2018. Can computational talent be detected? Predictive validity of the Computational Thinking Test. *International Journal of Child-Computer Interaction*. DOI: <http://dx.doi.org/10.1016/j.ijcci.2018.06.004>.
- [23] A. Rojas-López and F.J. García-Peñalvo. 2016. Personalized contents based on cognitive level of student's computational thinking for learning basic competencies of programming using an environment b-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. (ICPS: ACM International Conference Proceeding Series, New York, NY, USA: ACM, 2016, pp. 1139-1145.
- [24] A. Rojas-López. 2017. Intervention of three educational strategies for programming courses in higher education. *Education in the Knowledge Society* 18, 4, 21-34. DOI: <http://dx.doi.org/10.14201/eks20171842134>.
- [25] F.J. García-Peñalvo. 2013. Education in knowledge society: A new PhD programme approach. 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, 575-577. DOI: <http://dx.doi.org/10.1145 / 2536536.2536624>.
- [26] F.J. García-Peñalvo. 2014. Training in the knowledge society, a doctoral program with an interdisciplinary perspective. *Education in the Knowledge Society* 15, 1, 4-9.
- [27] F.J. García-Peñalvo. 2015. Engineering contributions to a Knowledge Society multicultural perspective. *IEEE Ibero-American Magazine of Learning Technologies (IEEE RITA)* 10, 1, 17-18. DOI: <http://dx.doi.org/10.1109/RITA.2015.2391371>.
- [28] F.J. García-Peñalvo. 2016. Presentation of the GRIAL research group and its main research lines and projects on March 2016. <https://goo.gl/dSZYv7>.
- [29] GRIAL Group. 2018. GRIAL Research Group Scientific Production Report (2011-2017). Version 2.0. Technical Report. GRIAL Research Group, University of Salamanca.



POST-PRINT