View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Gestion del Repositorio Documental de la Universidad de Salamanca

J. A. Martínez-Valdés, F. J. García-Peñalvo and J. A. Velázquez-Iturbide, "The role of basic mathematics concepts in programming teaching and learning," in TEEM'19 Proceedings of the Seventh International Conference on Technological Ecosystems for Enhancing Multiculturality (León, Spain, October 16th-18th, 2019), M. Á. Conde-González, F. J. Rodríguez-Sedano, C. Fernández-Llamas and F. J. García-Peñalvo, Eds. ICPS: ACM International Conference Proceedings Series, pp. 1046-1054, New York, NY, USA: ACM, 2019. doi: 10.1145/3362789.3362933.

The role of basic mathematics concepts in programming teaching and learning

José Alfredo Martínez-Valdés
Centro de Biotecnología Industrial
Servicio Nacional de Aprendizaje
763331 Palmira, Valle, Colombia
jamartinez6945@misena.edu.co

Francisco José García-Peñalvo
GRIAL Research Group
Research Institute for Educational Sciences, University of Salamanca
37008 Salamanca, Spain
fgarcia@usal.es

J. Ángel Velázquez-Iturbide
ETS Ingeniería Informática
Universidad Rey Juan Carlos
28933 Móstoles, Madrid, España
angel.velazquez@urjc.es

ABSTRACT

App Inventor is an online tool to create apps for mobile devices with Android operating system. With this tool, teachers can develop applications and video games that help students to learn fundamental concepts of mathematics through programming. The implementation of the integer division algorithm firstly in App Inventor, and later in C++ using successive subtractions required a detailed explanation of this algorithm. We have implemented an algorithm that uses the integer division theorem in response to a problem detected among mathematics teachers interested in teaching programming. This study involved 19 participants in 4 institutions in Palmira and Medellín (Colombia). The results indicated that the programming languages such as C++, Java and Python presented the erroneous results in the integer division when the divisor is a negative number, a similar situation to that was observed among the participants who presented quotients and incorrect residues in most cases. Using App Inventor in a programming course with teachers Maths allowed us to enrich the application with a video game for the student' self-evaluation in relation to the appropriation of the algorithm of the division of integer numbers.

GRIAL POST-PRINT

KEYWORDS

CS0; STEM Didactic; Sentiments Analysis; Abstraction; Math teaching, Integer Division Algorithm.

1. Introduction

Currently, there is a great concern regarding teaching and learning of programming in pre-university education[8–10, 27, 28]. However, in order to achieve a greater success, it is important to recognize the similarity between the algorithms of formal mathematics and their implementation in a programming language.

Due to the relative easy that App Inventor (AI2) presents for the development of mobile applications that can be immediately tested by students on their mobile devices; the authors decided to verify if AI2 can motivate students enough for them to decide to implement their apps developed in AI2 in native applications developed in a programming language such as Java for personal use. Another objective was to verify that low appropriation of the fundamental concepts of mathematics contributes to the students' apathy toward programming. The following research hypotheses were formulated:

H1. The transition of AI2 as a programming language prior to the use of Java in pre-university education requires the explanation of the fundamental concepts of mathematics and basic programming concepts[24].

H2. The development of applications initially in AI2 contributes to the Teachers Maths advance towards the development of native applications in formal languages of programming based on text.[12].

Providing an initial course of programming to people who have incomplete or inadequate mathematical training, and those who lack the ability to manipulate abstractions with ease, it becomes a challenge. These cognitive characteristics of the students directly affect the academic performance and the teaching - learning process in general, especially in abstract disciplines such as programming and mathematics. The teaching of programming has been approached from different angles. Generally, traditional approaches have as a side effect that most of students leave early due to lack of comprehension, notably diminishing the rates of graduation in the University, since such approaches usually do not consider the previous knowledge and the previous context that the students possess.[17, 29]

The main reason to teach mathematics through programming is that the student needs to rethink the mathematical concepts to translate them into a language that the computer understands, being able to favor the acquisition of strategies to solve problems. In addition, the knowledge acquired in programming can also be used to discuss classical concepts and math problems [3–5]

The teaching of programming supposes clarity on mathematical concepts underlying the solution of the problems to be solved, becoming the fundamental base for the teaching and learning of basic concepts of programming (types of data, variables, arithmetic operators, among others).

The integer numbers as a numerical set present properties of great importance for computer science and their study must include the integer division algorithm (IDA) [13].

Formal programming languages such as Java, C++ and Python have real division and integer division implemented. But the latter does not deliver the expected results in all cases Figure 1.

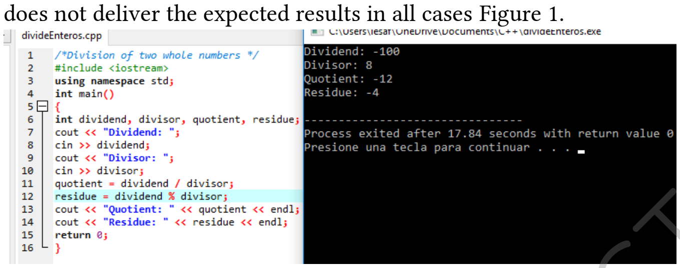

Figure 1: Coding and response of the algorithm of the entire division in C++

The structure of the paper is as follows. In the second section, we describe the research setting, including the participants, our research questions and measurement instruments. The third section presents the results of analyzing data gathered from different sources. Finally, sections 4 and 5 contain a discussion of the results and our conclusions, respectively.

2. Research setting

In this section, characterization of participants, the research goals and the instruments used are set.

2.1 Course Demography

19 teachers were interviewed, 7 through an online questionnaire and the others in printed format; 5 teachers in training from Medellín City, 4 teachers from Antonio Lizarazo Educational Institution from Palmira City, and 3 teachers from Sagrada Familia Educational Institution from Palmira City. Although many educational institutions were visited, Teacher's attitudes to fill the form was not the best. On other occasions, some of them expressed their support and collaboration, indicating they would take the questionnaire later. But, on returning to pick them up, you could tell they hadn't taken them.

After a few weeks of classes, we asked teachers to fill out an online questionnaire, in which they had to deliver their personal information. The questionnaire was composed of fourteen questions about:

- Institution to which they belong (optional)
- Personal data (gender, educational level in which he/she works, professional title and maximum educational attainment).

Another questions allowed us to know the level of appropriation of the algorithm of the division in integer numbers.

The results of this questionnaire are given in the third section.

The algorithm of the entire division.

Theorem: If a and b are integers with $b > 0$, then there are two integers, q and r, unique, such as $a = bq + r$, with $0 \leq r < b$. The numbers a, b, q and r are called, respectively, dividend, divisor, quotient and residue.

Corollary: If a and b are integers with $b \neq 0$, then there are two integers, q and r, unique, such that $a = bq + r$, where $0 \leq r < |b|$.

The remains are very interesting in Computational Algebra, both for the expression of a number in different numbering systems and for the applications of modular arithmetic, and, as seen in the theorem, they are positive numbers. It is because of this fact and because the division between negative numbers that is difficult for the student, since in some texts this theorem is restricted to the case of b, a positive number [20].

Table 4 shows that when b is less than zero, the situation got worse. Only 5% of participants correctly answered the second question, consequently that question have the worst results; perhaps because in the questionnaire it was the first question with a negative divisor.

One of the activities in mathematics education is the representation of fractions in the Cartesian plane, Thales Theorem is used for that purpose. However, the use of this theorem implies the division of a segment into equal parts, which enlarges the problem.

Determining the quotient and the residue provides a great help to represent the fraction that is generated from the numerator -10 and the denominator 3, or from the numerator 10 and the denominator -3.

2.2 Research Goals

We addressed two research goals:

RQ1. What is the Teacher's appropriation level about integer division algorithm?

RQ2. What is the role fundamental concepts of mathematics in the teaching and learning of basic programming concepts?

2.3 Materials and methods

The methodology used in the development of this research is presented in Table 1 and it is divided into phases so that the student can conceptualize the logical algorithmic path to find a solution.

Two questionnaires have been designed, one aimed at students in pre-university education and the other one to teachers in training or in service. It was decided to choose the sample with eighth and ninth-grade students because it was wanted to know if

POST-PRINT

they had studied the integer division algorithm. The questionnaire that was applied to the students of middle education showed disappointing results, then, it was applied to apprentices at the National Service of Learning (SENA, Spanish Acronym), with equally cast disappointing results, reason why it was decided to centralize the study exclusively to teachers and to apply the same questionnaire to them, differing from the students' questionnaires only in the demographic data section. The pretension of the questionnaire applied to teachers was to determine what was its integer division algorithm appropriation level to study the effect of the appropriation of basic concepts of mathematics in the teaching and learning of the programming.

The statement of the problem to be solved is the following: construct a program that, given two positive or negative integers as divider and divider entries respectively, this program shows as output two integers respectively quotient and residue that assist the algorithm of the division of integers according to the mathematical formulation.

This questionnaire consists of two pages, first, demographic data, also the mathematical algorithm formulation of the division of the whole numbers, and the second one, 9 questions in which the dividend and the divisor are presented so that those evaluated write the quotient and the residue[14]. For the application of this questionnaire, it has been considered appropriate to make a selection of teachers to have a varied sample, meaning, teachers with very different profiles in terms of how they see the teaching of Mathematics and its implementation; Among those, the teachers in charge of guiding this subject in the groups of students selected for the application of the tests are included. This questionnaire has been answered, done by 19 teachers of Highschool Education and Vocational Media of the Colombian educational system, which included teachers in training and teachers in service.

The compilation of information included teachers from Sagrada Familia Educational Institution and the Antonio Lizarazo Educational Institution from Palmira City, and teachers in training from Medellín City, Colombia.

With the information collected, the need of developing this algorithm in a programming language is observed. Initially, AI2 is used due to its characteristics of visual language based on blocks, its ability to provide visual information on the execution of computer programs [11, 18], to be multiplatform and due to the opportunity it offers to simulate applications on the computer or to install directly on mobile devices.

Table 1. Phases of the Methodology

Source: Own elaboration

Phase	Description	Objective
1	Problem Statement	In this phase, the problem is presented in a hasty manner so that students' comprehension of the problem is verifiable. This part is carried out with SENA apprentices in the Technology Program in Analysis and Development of Information Systems II Quarter of 2019
2	Analysis of the Solution	It opens an appropriate and timely manner so that students pose different solutions throughout brainstorming
3	Algorithm Statement	It is chosen one of the possible solutions according to the algorithm
4	Development of the Desktop Test	A desk test is done on paper to verify its validity and the achievement of the proposed objective
5	Refinement of the Algorithm	Time is allocated to analyze the characteristics of the algorithm and therefore optimize it
6	Implementation	The documentation is used as a way to store the logic solution for the problem, alternatives are sought to develop basic programming concepts (data type, variables, lists, loops, conditionals, logical operators, mathematical operators, etc.); the application is developed using AI2.
7	Fingering	The code in PSeInt, C++, and Java language is transcribed into the computer
8	Execution	It is run using the PSeInt IDE, the Java NetBeans IDE, and the C++ Language IDE called DevC++
9	Verification of Results	It is verified if the program delivers the expected results as established by the mathematical theory.

POST-PRINT GRIAL

3. Results

We present in this section the results of our research. First, we present the demographic characteristics of the enrollment. Then, we present the findings related to the two research questions.

3.1 Course Demography

The results of the initial questionnaire are shown in Table 2.

Table 2. Different features of enrolled teachers

Source: Own elaboration

Factor	Value	# teachers	%
Maximum Educational Level Achieved	Specialization	3	16
	Master's	5	26
	Superior teacher	1	5
	Professional	10	53
Gender	Man	14	74
	Woman	5	26
Performance level	Basic high school	12	63
	Vocational Media	3	16
	Elementary school	3	16
	Teacher in training	1	5
Professional Title	Bachelor in mathematics	6	32
	Professional without bachelor	6	32
	Superior teacher	1	5
	Bachelor of Basic Education (With an emphasis in mathematics)	6	32

Note that most Maths Teachers are men, and have undergraduate studies.

3.2 Assessment of the appropriation level the integer division algorithm with math teachers

Table 3 shows the correct answers to the nine problems related to the integer numbers division algorithm that was presented to the participants in the questionnaire.

Table 3. Expected responses in the questionnaire

Source: Own elaboration

Question	a	b	q	r	bq	a = bq + r, where $0 \leq r < \|b\|$
1	-102	13	-8	2	-104	-102 = -104 + 2
2	-102	-13	8	2	-104	-102 = -104 + 2
3	-91	101	-1	10	-101	-91 = -101 + 10
4	100	-8	-12	4	96	100 = 96 + 4
5	-100	-8	13	4	-104	-100 = -104 + 4
6	102	-13	-7	11	91	102 = 91 + 11
7	91	-101	0	91	0	91 = 0 + 91
8	-91	-101	1	10	-101	-91 = -101 + 10
9	-100	8	-13	4	-104	-100 = -104 + 4

3.2.1 Responses sent by the participant teachers

Table 4 shows the diversity of responses that participants chose for the first question of the questionnaire. It is observed that at least one of the participants did not take into account that integer numbers do not have decimals and that only four out of the nineteen participants answered correctly this question.

Table 4. Responses sent by the participant teachers (Q1, N = 19)

Source: Own elaboration

quotient	% of answers	residual	% of answers
-7	63%	-11	37%
-7.846115384	5%	8	5%
-9	5%	11	26%
-8	21%	84	5%
-3	5%	2	21%
		9	5%

Table 5 shows the diversity of responses that participants chose for the second question of the questionnaire. It is observed that at least one of the participants did not take into account that integer numbers do not have decimals and that only one out of nineteen participants answered this question correctly.

GRIAL POST-PRINT

Table 5. Responses sent by the participant teachers (Q2, N = 19)

Source: Own elaboration

quotient	% of answers	residual	% of answers
-7	63%	-11	32%
-7.846115384	5%	8	5%
-9	5%	11	47%
7	21%	84	5%
8	5%	9	5%
		2	5%

Table 6 shows the diversity of responses that participants chose for the third question of the questionnaire. It is observed that at least one of the participants did not take into account that integer numbers do not have decimals and that only two out of the nineteen participants answered this question correctly.

Table 6. Responses sent by the participant teachers (Q3, N = 19)

Source: Own elaboration

quotient	% of answers	residual	% of answers
7	21%	11	16%
7.846115384	5%	8	5%
9	5%	-11	5%
8	11%	84	5%
-0.9	21%	2	11%
-0.8	5%	-0.1	5%
-1	11%	1	16%
0	11%	10	11%
NR	5%	-91	11%
12	5%	NR	11%

Table 7 shows the diversity of responses that participants chose for the fourth question of the questionnaire. It is observed that two of the participants did not take into account that the integer numbers do not have decimals and that approximately half of the participants answered this question correctly.

Table 7. Responses sent by the participant teachers (Q4, N = 19)

Source: Own elaboration

quotient	% of answers	residual	% of answers
-0.9	21%	-1	5%
-0.90099	5%	101	5%
0.9	5%	0.1	11%
-1	5%	1	11%
0	5%	90	5%
-12	47%	91	5%
-12.5	11%	4	42%
		0	11%
		7	5%

Table 8 shows the diversity of responses that the participants chose for the fifth question of the questionnaire. It is observed that two of the participants did not take into account that the integer numbers do not have decimals and that it is more difficult to find the residue correctly than the quotient due to its greater diversity in the answers presented.

Table 8. Responses sent by the participant teachers (Q5, N = 19)

Source: Own elaboration

quotient	% of answers	residual	% of answers
-0.9	26%	-11	5%
-0.90099	5%	8	5%
1	5%	11	11%
-1	5%	-1	5%
12	37%	84	5%
12.5	11%	2	11%
13	11%	4	26%
		0	11%
		-4	21%

Table 9 shows the diversity of responses that the participants chose for the sixth question of the questionnaire. It is observed that two of the participants did not take into account that integer numbers do not have decimals and that this one is the question with the highest percentage of correct answers.

GRIAL POST-PRINT

Table 9. Responses sent by the participant teachers (Q6, N = 19)

quotient	% of answers	residual	% of answers
0.9	26%	-11	5%
0.90099	5%	8	5%
1	11%	11	79%
-7	47%	84	5%
-3	5%	9	5%
7	5%		

Table 10 shows that the seventh question presents a lot of variety in the answers and that less than a quarter of the participants chose the correct answer to this question.

Table 10. Responses sent by the participant teachers (Q7, N = 19)

Source: Own elaboration

quotient	% of answers	residual	% of answers
-12	26%	11	16%
-12.5	11%	8	5%
-11	5%	-11	5%
-0.9	21%	84	5%
-0.81	5%	2	11%
0	21%	0.9	5%
NR	5%	1	21%
0.9	5%	91	21%
		NR	5%
		0.01	5%

Table 11 shows that the eighth question presents a lot of variety in the answers and that about 90% of the participants responded to this question incorrectly.

Table 11. Responses sent by the participant teachers (Q8, N = 19)

Source: Own elaboration

quotient	% of answers	residual	% of answers
-12	16%	-1	5%
-12.5	11%	101	5%
-11	5%	0.1	16%
-13	11%	1	32%
0.9	26%	90	5%
0.81	5%	91	5%
1	11%	10	11%
0	11%	-91	11%
NR	5%	NR	5%
		-0.01	5%

Table 12 shows the different answers to the last question of the questionnaire. It can be seen that, although there is a higher percentage of participants who chose the residue correctly, there are many mistakes when calculating the quotient, less than half of the participants answered correctly to this question.

Table 12. Responses sent by the participant teachers (Q9, N = 19)

Source: Own elaboration

quotient	% of answers	residual	% of answers
12	16%	8	5%
12.5	11%	0	16%
11	5%	4	42%
13	11%	-12	5%
-12	37%	5	5%
-12.5	11%	-4	26%
-13	11%		

3.2.2 Implementation

We have implemented the algorithm in PSeInt, in C++ and Java and the expected results were consistent with the theorem. (see Table 13).

POST-PRINT
GRIAL

Table 13. Different resources for the implementation of the algorithm

Code	Description	Comment

`#include <wchar.h>`
`#include <locale.h>`
`#include <stdlib.h>`
`#include<iostream>`
`#include <math.h>`
`#include <stdlib.h>`
`using namespace std;`
 	
`int b1, counter, d, divisor, q, q1, r;`
`bool one, two, three, four;`
	Imported libraries Definition of variables	Integer data types to calculate quotient and remainder Boolean data types for comparisons

`cout << "\nPlease, enter the dividend" << endl;`
`cin >> d;`
`cout << "\nPlease, enter the divisor" << endl;`
`cin >> divisor;`
`cout << "\n" << endl;`
`b1 = d;`
`q1 = divisor;`
`counter = 0;`
`one = ((d>=0) && (divisor>0));`
`two = ((d<0) && (divisor<0));`
`three = ((d>0) && (divisor<0));`
`four = ((d<0) && (divisor>0));`
 	
`if ((one==true)) {`
`d = (d-divisor);`
`while ((d>=0)) {`
`counter = counter+1;`
`d = (d-divisor);`
`}`
`q = counter;`
`r = d+divisor;`
`cout<<"";Output();//out(b1, q1, q, r)`
`} else {`
`if ((two==true)) {`
	The process for calculating the quotient and remainder based on the signs of dividend and divisor	We have used successive subtractions because the operators / and% (for division and module) are not allowed.

`Please, enter the dividend`
`-67`
`Please, enter the divisor`
`13`
`the division algorithm for integers states that`
`r such that 0 < r < b and a = bq + r.`
`The numbers q and r should be thought of as the`
`he remainder r is non-negative and is always le`
`Dividend is: -67`
`divisor is: 13`
`quotient is: -6`
`residue is: 11`
`It is verified that: -67 = 13*(-6)+ 11`
	in C++	The results in C++ are the expected ones according to the mathematical formulation of the algorithm of the division of integers.

`BlueJ: Ventana de Terminal - DIVALGO`
`Opciones`
`Please enter the dividend`
`-100`
`Please enter the divisor`
`8`
`Dividend is: -100`
`Divisor is: 8`
`quotient is: -13`
`residue is: 4`
`It is verified that: -100 = 8*(-13)+ 4`
	in Java	The results in Java are the expected ones according to the mathematical formulation of the algorithm of the division of integers.

One of the aspects that increase the difficulty of learning to program is that, in general, programming courses include a general-purpose programming language, such as Java, C++ or Python, an integrated development environment (IDE) for the programming language chosen. This situation makes those students to be dedicated to learn the syntax of the program, and the efficient administration of the IDE and it does not make what should be the most important thing, it means, the resolution of the problem[21]. The use of a visual programming language (VPL) such as AI2 allows focusing on solving the problem, also allowing the development of artistic skills in students at an early age. (see figures 1 and 2) [12, 25]

Figure 2 shows that AI2 offers Math 'Teachers the opportunity to diversify the contexts application of the concepts taught in school of mathematics in situations that may be more attractive to students and increase the motivation to study, such as the development of videogames through this diversity of contexts the teacher can use a rapid implementation to address diversity itself and uphold inclusion.

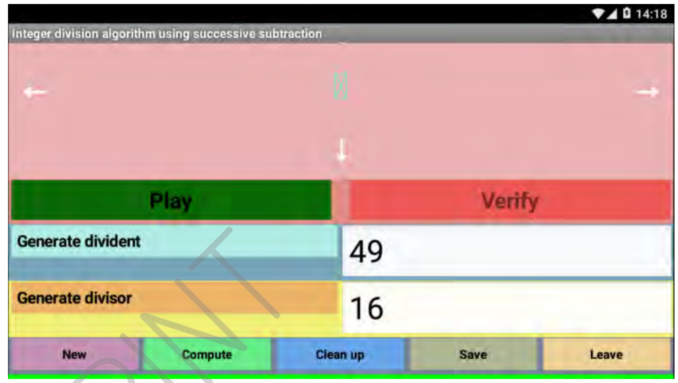

Figure 2: Game developed in AI2

Figure 3 shows that AI2 offers the opportunity to enlarge the didactic by having the possibility of teaching some of the concepts of mathematics, where it can be taught and reinforced through basic programming concepts. Implementing the algorithm of the division of integers in AI2 allows us to explain the concept of variables, logical operators, control structures, selection of sentences, and repetition of sentences; to generate the divisors of a number, to write them in a list and to publish the multiplication tables depending on the dividend and the divisor.

POST PRINT

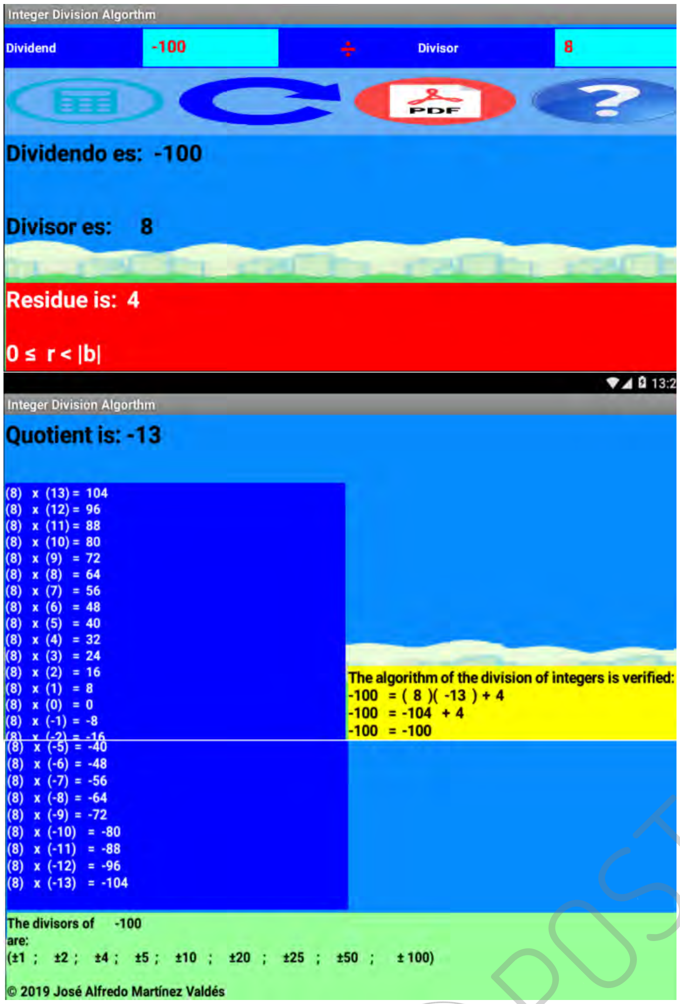

Figure 3: App developed in AI2

4. Discussion and conclusions

We presented the AI2 in a teacher training course at CS0 high school. The results are very satisfactory, the use of AI2 with teachers of mathematics has allowed them to implement the integer division algorithm by means of successive subtractions. Now, teachers have the possibility to extend the applications and video games use and developing (Sánchez-Prieto et al.) to help them in their task Maths Teacher with a correct didactic transposition [6, 7, 18, 19].

AI2 is an easy-to-use, multiplatform programming language that does not need to be installed, it works on the web, therefore, it guarantees portability and easy access, it does not require compilers or the installation of additional software[1].

AI2 is indicated for students to focus on solving problems rather than learning the syntax of a language [22]. The motivation of the student stands out even more than other similar applications such as Scratch or Alice, "because of its portable nature and its usefulness", since it is based on the use of mobile phones, something important in the students' lives.

Like Alice and Scratch, AI2 has the disadvantage that it is not used in the industry, for this reason, these programming languages can be unattractive to students with some programming experience[15]. Therefore, it is recommended to attract novices to programming[16] and use App Inventor as a language prior to the use of formal programming languages based on text[1, 2, 21, 22, 26].

Our work resembles that of [21] insofar as we intend to show that AI2 allows a non-programmer teacher to create personalized applications that are useful for the students' learning process in areas such as mathematics [12].

This research is part of a larger project that seeks to provide input for a specific teaching of programming based on the analysis of the students' opinions during the teaching and learning event.

ACKNOWLEDGMENTS

This work was supported by GRIAL Research Group of Research Institute for Educational and by Doctoral Program in Computer Engineering from University of Salamanca.

GRIAL POST-PRINT

REFERENCES

[1] Ali, A. et al. 2017. Comparing novice programing environments for use in secondary education: App Inventor for Android vs. Alice. ACM International Conference Proceeding Series. 10, 1/2 (2017), 44. DOI:https://doi.org/10.1504/ijtel.2018.088333.

[2] Ali, A. and T Smith, D. 2017. Teaching an Introductory Programming Language in a General Education Course. Journal of Information Technology Education: Innovations in Practice. 13, (2017), 057–067. DOI:https://doi.org/10.28945/1992.

[3] Barcelos, T.S. et al. 2018. Mathematics learning through computational thinking activities: A systematic literature review. Journal of Universal Computer Science. 24, 7 (2018), 815–845.

[4] Du Boulay, J.B. 1980. Teaching teachers mathematics through programming. International Journal of Mathematical Education in Science and Technology. 11, 3 (1980), 347–360. DOI:https://doi.org/10.1080/0020739800110306.

[5] Briz, Á. and Serrano, Á. 2018. Aprendizaje de las matemáticas a través del lenguaje de programación R en Educación Secundaria. Educacion Matematica. 30, 1 (2018), 133–162. DOI:https://doi.org/10.24844/EM3001.05.

[6] Cruzeiro, U. and Brousseau, G. 2015. Trigonometry in triangle rectangle: interactions in the classroom the perspective of didactic theory of situations. 1, (2015). DOI:https://doi.org/10.15628/holos.2015.1616.

[7] Fayad, A.M. 2017. Elementos estructurantes de la Didáctica de la Informática Structural elements of the Teaching of Informatics. 14, (2017), 100–110.

[8] García-Peñalvo, Francisco J.; Mendes, J.A. et al. 2018. Exploring the computational thinking effects in pre-university education. Computers in Human Behavior. 80, (Dec. 2018), 407–411. DOI:https://doi.org/10.1016/j.chb.2017.12.005.

[9] Garcia-Penalvo, F.J. 2018. Computational thinking. IEEE Revista Iberoamericana de Tecnologías del Aprendizaje (IEEE RITA) 13, 1, 17-19. (2018). DOI:https://doi.org/10.1109/RITA.2018.2809939.

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

[11] Insuasti, J. 2016. Problemas de enseñanza y aprendizaje. Revista educación y desarrollo social. 10, 2 (2016), 234–246. DOI:https://doi.org/10/18359/reds.1701.

[12] Ke, F. 2014. An implementation of design-based learning through creating educational computer games: A case study on mathematics learning during design and computing. Computers & Education. 73, (Apr. 2014), 26–39. DOI:https://doi.org/10.1016/j.compedu.2013.12.010.

[13] Leijen, D. 2001. Division and Modulus for Computer Scientists. University of Utrecht. http://www. cs. uu. nl/~ daan/lvm. .... 2 (2001), 1–7.

[14] Martínez-Valdés, J.-A. 2019. Integer Division Algorithm: Questionnaire for teachers.

[15] Martínez-Valdés, J.A. et al. 2017. A (Relatively) Unsatisfactory Experience of Use of Scratch in CS1. Fifth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'17) (Cádiz, Spain, October 18-20, 2017) J.M. Dodero, M.S. Ibarra Sáiz and I. Ruiz Rube Eds. ACM, New York, NY, USA,

Article 8. DOI:10.1145/3144826.3145356. ((Cádiz, Spain, October 18-20, 2017) (Article 8). New York, NY, USA:, 2017).
[16] Martínez-Valdés, J.A. and Martínez-Ijají, N.A. 2018. An experience with the App Inventor in CS0 for the development of the STEM didactics. Proceedings TEEM'18. Sixth International Conference on Technological Ecosystems for Enhancing Multiculturality (Salamanca, Spain, October 24th-26th, 2018), F.J. García-Peñalvo Ed. ACM, New York, NY, USA, 51-56. DOI:10.1145/328417 (2018).
[17] Martínez López, P. et al. 2012. El nombre verdadero de la programación. 10° Simposio sobre la Sociedad de la Información - 41 Jornadas Argentinas de Investigación Operativa,. (2012), 1–23.
[18] Miños, A. et al. 2019. Enfoques metodológicos para la enseñanza de la programación en la escuela secundaria Resumen. Información tecnológica. 1, May (2019), 100–110. DOI:https://doi.org/10.15628/holos.2015.1616.
[19] Miños, A. 2016. Uso didáctico de estrategias inductivas en un curso introductorio de programación estructurada. Información tecnológica. 1, 1 (2016), 25.
[20] Ordónez, L;Ordóñez, C; Contreras, A. and Ordóñez, C.; Ordóñez, L.; Contreras, A. 2015. La divisibilidad en manuales para estudiantes de ingeniería informática. (2015), 431–440.
[21] Ortega García, A. et al. 2018. Using App Inventor for creating apps to support m-learning experiences: A case study. Computer Applications in Engineering Education. 26, 3 (2018), 431–448. DOI:https://doi.org/10.1002/cae.21895.
[22] Papadakis, S. and Orfanakis, V. 2017. Comparing novice programing environments for use in secondary education: App Inventor for Android vs. Alice. International Journal of Technology Enhanced Learning. 10, 1/2 (2017), 44. DOI:https://doi.org/10.1504/ijtel.2018.088333.
[23] Sánchez-Prieto, J.C. et al. Motivación e innovación: Aceptación de tecnologías móviles en los maestros en formación. Revista Iberoamericana de Educación a Distancia 20, 2, 273-292. DOI: 10.5944/ried.20.2.17700.
[24] Toh, P.C. et al. 2016. Examples in the Teaching of Mathematics: Teachers' Perceptions. Developing 21st Century Competencies in the Mathematics Classroom. (2016), 189–211. DOI:https://doi.org/10.1142/9789813143623_0011.
[25] Tsai, C.Y. 2019. Improving students' understanding of basic programming concepts through visual programming language: The role of self-efficacy. Computers in Human Behavior. 95, May 2018 (2019), 224–232. DOI:https://doi.org/10.1016/j.chb.2018.11.038.
[26] Tsai, C.Y. 2019. Improving students' understanding of basic programming concepts through visual programming language: The role of self-efficacy. Computers in Human Behavior. 95, November (2019), 224–232. DOI:https://doi.org/10.1016/j.chb.2018.11.038.
[27] Velázquez-Iturbide, J. Ángel.; Bahamonde, A; Dabic, S; Escalona, M. J.; Feito, F; Fernández Cabaleiro, S; Ferrero Martín, B; Garay Vitoria, N; García, J. C.; Rios Zapata, M.;García Borgoñón, L.; García Molina, J.; Larraza Mendiluze, E. 2018. Informe del Grupo de Trabajo SCIE/CODDII sobre la enseñanza preuniversitaria de la informática. Sociedad Científica Informática de España, Conferencia de Decanos y Directores de Ingeniería Informática.
[28] Velázquez-Iturbide, J.Á. 2018. Report of the Spanish Computing Scientific Society on Computing Education in Pre-University Stages. Pre-University Stages. In Proceedings TEEM'18. Sixth International Conference on Technological Ecosystems for Enhancing Multiculturality (Salamanca, Spain, October 24th-26th, 2018), F.J. García-Peñalvo Ed. ACM, New York, NY, USA, 2-7. DOI:10.1145/3284179. (2018), 2–7.
[29] Zabala, G. et al. 2013. Una propuesta de enseñanza de programación en escuela media mediante el desarrollo de videojuegos con Etoys. (2013).

POST-PRINT
GRIAL