Teaching the sequential programming concept using a robotic arm in an interactive museum

Federico Botella
I.U. Centro de Investigación Operativa
Universidad Miguel Hernández
Elche, España
federico@umh.es

Antonio Peñalver
I.U. Centro de Investigación Operativa
Universidad Miguel Hernández
Elche, España
a.penalver@umh.es

Manuel Quesada-Martínez
I.U. Centro de Investigación Operativa
Universidad Miguel Hernández
Elche, España
mquesada@umh.es

Fulgencio Bermejo
Dep. Estadística, Mat. e Informática
Universidad Miguel Hernández
Elche, España
fbermejo@goumh.umh.es

Fernando Borrás
Dep. Estadística, Mat. e Informática
Universidad Miguel Hernández
Elche, España
f.borras@umh.es

ABSTRACT

Technology and Computer Science are increasingly present in today's education and teaching programming is not only restricted to students interested in Science, Technology, Engineering and Maths (STEM) disciplines, as computational thinking is useful in many day-to-day problems. In this paper we study how students of high school can put in practice transversal concepts by learning the sequential programming concept. We analyze their learning process by asking them to code a simple program that solves a concrete problem: perform simple and immersive tasks using a physical robot in an interactive museum. The experiment offers us some results that should be confirmed with more participants, but it seems that the ages from 13 to 15 years old are crucial to gain knowledge and skills to apply concepts of their studies on using sequential programming to interact with a robotic arm.

CCS CONCEPTS

• Human-centered computing → Human computer interaction (HCI) → HCI design and evaluation methods → Usability testing

KEYWORDS

learning efficiency; interactive module; science museum; robot arm; sequential programming

Interacción 2019, June 25–28, 2019, Donostia, Gipuzkoa, Spain

ACM ISBN 978-1-4503-7176-6/19/06…$15.00
https://doi.org/10.1145/3335595.3336289

ACM Reference format:

Federico Botella, Antonio Peñalver, Manuel Quesada-Martínez, Fulgencio Bermejo and Fernando Borrás. 2019. Teaching the sequential programming concept using a robotic arm in an interactive museum. In *XX International Conference on Human Computer Interaction (Interacción 2019)*, June 25–28, 2019, Donostia, Gipuzkoa, Spain. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3335595.3336289

1 Introduction

The use of digital technology in education is a truth. Nowadays, teaching code is not only restricted to students interested in science, technology, engineering and math disciplines (STEM), but also it enables to any student learning skills beyond the coding itself [14] such as computational thinking that is present in many daily activities and problems.

At present, high school students in Spain (ages 12 to 16) usually visit museums and science and technology centers, hereinafter "science museums". In these museums, different interactive modules are exposed for showing its usefulness to teach some science branches or technological principles [12, 13]. In this work, we perform an experiment in the Science Interactive and Didactic Museum of the Vega Baja del Segura of the Comunitat Valenciana, hereinafter "MUDIC", located in Orihuela (Spain) [11]. MUDIC is daily visited mainly by students of High School, but also by Baccalaureate and Vocational Training students and associations of adults. The museum has several thematic rooms by branches of science.

The goal of our experiment is to analyze the use of the interactive robotic arm module of the MUDIC to teach students the concept of sequential programming and to measure their efficiency in this interactive module. Sequential programming consists of a series of instructions executed following the same sequence and the same timing, which in the context of the robotic arm are used to move an object. We asked the participants to program the robot by coding the sequence of movements needed to move an object from one place to another.

2 Related work

Computers are utilized by teachers to guide students in the search for knowledge, so computers can provide many learning opportunities. The didactic and constructive benefits of using technologies for teaching is discussed in [16], and a set of qualitative research findings is used to perform to set out the worth or value of digital technology in education in [23]. A recent study [5] performed in the second cycle of early childhood education during 2016-2017 academic period allowed students to solve programming challenges, and both teachers and students shown their favorable acceptance of the technical, pedagogical and social aspects of the experience. The effects of computer-based teaching in secondary schools are developing positive attitudes and reducing substantially the amount of time that students needed for learning [8].

Scratch is one of the prominent examples of tools for teaching and promote the computational thinking (CT) between young students. The Scratch success and the vast number of projects generated with it has attracted attention of the scientific community as interactive learning environments [10]. Scratch programming interface have an attractive graphical user interfaces (GUIs) based on draggable blocks, so the interaction and the way of coding is close to a traditional puzzle, making coding up to all kind of students. The use of friendly GUIs for learning programming is a common practice in interactive learning environments, as they help students to acquire some abstract competencies and they increase the usability of traditional tools [6]. Recently, informal learning environment based on friendly GUIs are combined with robots to create immersive experiences where students observe the consequences of their algorithms on reality [3, 5].

The study of museums as educational and learning experiences has been done by different authors [4, 12, 13]. Interactive modules have been used for teaching a wide variety of problems and disciplines in the context of a science museum. For instance, [9] visualizes the global distribution of phytoplankton across the time in an interactive module. In recent years, virtual and augmented reality experiences in museums have created novel immersive situations for visitors [7]. The evaluation of interactive modules comprises different scopes related with them. For example, in [1] the authors focus their efforts on identify common pitfalls of designing exhibits and they point out the importance and necessity of design principles for research, prototyping, and evaluation (formative and summative). They highlighted and discussed four important aspects of the learning environment: immediate learning ability, physical interactivity, conceptual coherence, and diversity of learning modes. Another important aspect to evaluate in interactive modules is the user experience of the visitors. In [15] a hybrid User Experience (UX) modelling framework is proposed, and they state that UX quality is the media quality consumed at a particular touchpoint. In the context of our robotic arm interactive module, the UX should analysis the UX of students with the GUIs for programming the robot, and with the robot as an immersive tool. In [2] the authors apply UX principles and scholarship to museum design and provide them with tools to evaluate the effectiveness of existing participatory elements.

Previous work evaluates the infrastructure of the museum and the role that interactive modules plays in the visitor's experience. However, none of them evaluates the effectiveness of the interactive modules in the learning process. In this paper, we want to study how high school students can put in practice previous concepts they had already learned to solve a concrete problem using sequential programming: perform simple tasks with a robotic arm.

3 Methods

The Interactive Robotic Arm module in MUDIC is a computer-controlled robot arm. The main objective of the interactive module is to teach the concept of sequential programming as an introduction to the world of programming. The robot arm has six degrees of freedom: rotation of the base, arm, forearm, wrist rotation, elevation of the wrist and claw. These operations let users to learn and practice the concept of sequential programming by recording a sequence of sentences that can be reproduced autonomously by the robot arm and repeated as many times as necessary. For example, we propose a simple sequence that consist of moving the robotic arm to a position, closing the claw to hold an object, moving the object to a different position and releasing it. The module provides a GUI executed on a PC, from which the user can control the movements, and therefore the position of the robotic arm by selecting a series of menu options on the screen. After getting the right position of the robotic arm, the user must press another button to store the current position. In this way, the sequence of program steps is generated.

3.1 Target audience of the study

Our study focused on students of ESO (Compulsory Secondary Education). They were randomly recruited from different educational centers that visited the MUDIC during the months of February and March of 2018. The experiment involved 26 students (16 boys and 10 girls), with an average age of 14,5 years (13 to 16 years old), from five different educational centers and from two educational levels (2nd and 4th level). All of the participants signed a consent form for participating in the experiment. It is important to note that in 2nd level, programming is not included in the curriculum, but in 4th level students already deal with basic concepts of coding languages and with the computer as control elements of robotic systems.

3.2 Experiment description

Figure 1 shows the robotic arm (Lynxmotion RIOS SSC-32) located in the MUDIC which is controlled by a card connected to a PC with proprietary control.

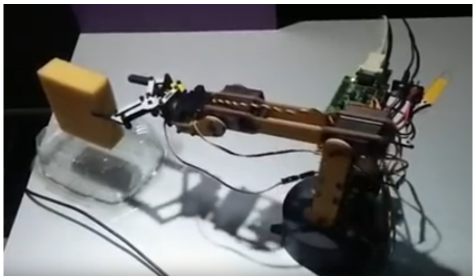

Figure 1: Robotic arm located in the MUDIC museum

As mentioned, some students could have no previous knowledge of programming, so we elaborated a 17 pages manual that contains instructions to learn some basic programming concepts, makes the previous configuration of the system and fully describes the steps to complete the proposed tasks to move an object from an initial position to another. Moreover, the manual explains other aspects like the GUI to be used, the type of coordinates for programming, the different parts of the robotic arm and the user identification.

We asked all participants to reply a brief pre-test questionnaire of three questions: (PT-Q1) Have you ever programmed any device? (Yes/No); (PT-Q2) Would you ever like to work in the field of Technology or Engineering? (Yes/No), (PT-Q3) What qualification have you obtained in Technology? (1 to 5). During the test, the participants were shown the purpose of the tasks to be performed by the movements of the robotic arm with a program that worked correctly. After that, the robotic arm was calibrated in an unfavorable position; then the students were indicated to use the manual, as most of them was the first time they programed a robotic arm. All the participants were asked to individually program the movements by recording it into a file, so that these five tasks were performed: (1) grab the object; (2) lift the object; (3) rotate the robot arm; (4) drop the object; and (5) check that the robot has been correctly programmed. Before starting the test, the robot was in the same unfavorable position to perform the task of grabbing the object, requiring several movements before reaching the grip position of the object. During the experiment no answers were replied. For any help, we suggested them to read the manual.

We measured the time needed by each student to accomplish tasks 1 to 5. It should be pointed out that students could read the manual as much times as they needed, however before performing task1, reading the manual was compulsory to grab the object. So, this first reading of the manual was included as a part of task1. We decided to considered two aggregated times: (tI) the time needed by participants for understanding the operation of the robot through the reading of the manual and perform the first task grabbing the object, and (tF) the time needed to perform tasks 2-3-4-5.

4 Results and discussion

The analysis of PT-Q1 answers revealed that only 2 of 10 girls had programmed (20.0%) whereas only 3 of 16 boys answered that they had programmed before (18.8%). In both cases, both boys and girls had never programmed a device before in 80%. The analysis of PT-Q2 answers revealed that 12 of 16 boys would like to work in a job related to technology or engineering (75%) whereas only 3 of 10 girls answered affirmatively, that is, a 30% of girls. These preliminary results were in line with the gender gap that girls do not want to work in jobs related to engineering or technology. The average qualification of the participants was 3.9 (PT-Q3).

Table 1 presents the average time in seconds (tI(s)) needed by girls (F) and boys (M) to perform the first part of the experiment, and the average time in seconds (tF(s)) needed by girls (F) and boys (M) to finish the rest of tasks. Virtually, both girls and boys needed the same average time, tI(s), to achieve the first task (around 7:30 min) and the same average time, tF(s) to complete the rest of tasks (around 2:20 min). The Student's T-test shows no significant differences in times between girls and boys to complete task1 neither to complete the rest of tasks as $p>0.05$ in both cases (see Table 2).

	Group	N	Mean	SD	SE
tI(s)	F	10	448.100	194.458	61.493
	M	16	446.125	161.206	40.302
tF(s)	F	10	130.000	65.818	20.813
	M	16	143.875	112.492	28.123

Table 1: Average times (in sec.) to perform task1 (tI(s)) and the rest of tasks (tF(s)) by girls(F) and boys(M).

However, the analysis of tI(s) revealed that girls and boys of 2nd level needed only one minute more of average than students of 4th level. While girls of 2nd level needed near one minute more to perform the task1 than girls of 4th level, boys of 2nd level needed near three minutes more than boys of 4th level. It seems that at these ages, girls achieve more reading skills needed for task1 than the boys (as this task includes the reading of the manual). In Table 3 we can find the average times (tI(s) and tF(s) in seconds) needed by students of 2nd level and 4th level, and the average time (tT(s)) to complete all the tasks (1-2-3-4-5).

Independent Samples T-Test

	t	df	p
tI(s)	0.028	24.000	0.978
tF(s)	-0.353	24.000	0.728

Table 2: Student's T-Test for times to perform task1 and to complete the rest of tasks by girls and boys.

Group Descriptive

	Group	N	Mean	SD	SE
tI(s)	2ESO	13	512.154	144.305	40.023
	4ESO	13	381.615	175.479	48.669

Group Descriptive

	Group	N	Mean	SD	SE
tF(s)	2ESO	13	165.385	112.321	31.152
	4ESO	13	111.692	70.626	19.588
tT(s)	2ESO	13	677.538	153.086	42.458
	4ESO	13	493.308	156.984	43.539

Table 3: Average times (in sec.) to perform: task1 (tI(s)), the rest of task (tF(s)) and all the experiment (tT(s)).

Now, we can find significant differences in times needed to achieve the task1 by students of 2^{nd} level (8:32 min) vs students of 4^{th} level (6:22 min). In Table 4 we can see a meaningful time difference with p=0.049 for time needed to complete task1 by students of 2^{nd} level vs students of 4^{th} level: the students of 4^{th} level needed 2:11 min less than students of 2^{nd} level to achieve task1. But also, a meaningful time difference in time needed to perform all the experiment, with p=0.006, by students of 2^{nd} level (11:17 min) and by students of 4^{th} level (8:13 min). It can be concluded than students of 4^{th} level needed near three minutes less than 2^{nd} level to perform all the tasks. It seems than knowledge and skills acquired in these two years are crucial to this significant time reduction.

Independent Samples T-Test

	t	df	p
tI(s)	2.072	24.000	0.049
tF(s)	1.459	24.000	0.158
tT	3.029	24.000	0.006

Table 4: Student's T-Test for times to perform task1, to complete the rest of tasks and to complete all the experiment by students of 2^{nd} ESO vs students of 4^{th} ESO.

5 Conclusions

Teaching computer science and teaching interacting with robots is always a difficult task, not only for young students of high schools but also for university students. To understand how to program a simple task to be performed by a robotic arm requires previous knowledge about concepts of programming, concepts of mathematics and robotics. In this paper we have conducted an experimental study of how students of secondary education can apply their knowledge to solve a concrete problem: perform simple tasks with a robotic arm by introducing them in the world of sequential programming. The experiment revealed that students of 4^{th} level of secondary education are more skilled and more effective than students of 2^{nd} level to interact with a robotic arm and apply concepts of their studies to sequential programming. Although we only could recruit a few participants to conduct this experiment, the results seem to be encouraging and we want to repeat it in near future, recruiting more students.

ACKNOWLEDGMENTS

Thank you to all participants, and especially the Director and instructors of the MUDIC of the Miguel Hernández University of Elche for the help we received to conduct the experiment.

REFERENCES

[1] Allen, S. 2004. Designs for learning: Studying science museum exhibits that do more than entertain. Science Education. 88, S1 (2004), 17–33.
[2] Babak, L. and Gibbons, J. 2018. UX Approaches for Museum Contexts. Proceedings of the 36th ACM International Conference on the Design of Communication - SIGDOC '18 (New York, New York, USA, 2018), 1–6.
[3] Barker, B.S. and Ansorge, J. 2007. Robotics as Means to Increase Achievement Scores in an Informal Learning Environment. Journal of Research on Technology in Education. 39, 3 (2007), 229–243.
[4] Council, N.R. and others 2009. Learning Science in Informal Environments. National Academies Press.
[5] González, Y.A.C. and Muñoz-Repiso, A.G.-V. 2018. A robotics-based approach to foster programming skills and computational thinking. Proceedings of the 6th International Conference on Technological Ecosystems for Enhancing Multiculturality (NY, USA, 2018), 41–45.
[6] Green, T.R.G. and Petre, M. 1996. Usability Analysis of Visual Programming Environments: A 'Cognitive Dimensions' Framework. Journal of Visual Languages & Computing. 7, 2 (1996), 131–174.
[7] Jung, T., tom Dieck, M.C., Lee, H. and Chung, N. 2016. Effects of Virtual Reality and Augmented Reality on Visitor Experiences in Museum. Information and Communication Technologies in Tourism 2016. Springer International Publishing. 621–635.
[8] Kulik, J.A., Bangert, R.L. and Williams, G.W. 1983. Effects of computer-based teaching on secondary school students. Journal of Educational psychology. 75, 1 (1983), 19.
[9] Ma, J., Liao, I., Kwan-Liu Ma and Frazier, J. 2012. Living Liquid: Design and Evaluation of an Exploratory Visualization Tool for Museum Visitors. IEEE Transactions on Visualization and Computer Graphics. 18, 12 (2012), 2799–2808.
[10] Marcelino, M.J., Pessoa, T., Vieira, C., Salvador, T. and Mendes, A.J. 2018. Learning Computational Thinking and scratch at distance. Computers in Human Behavior. 80, (2018), 470–477.
[11] Museo Didáctico e Interactivo de Ciencias: http://www.mudic.es.
[12] Nieves, M., Abellán, G. and Carnicer, J. 2009. Experiencias de química en un museo didáctico e interactivo de Ciencias. Anales de Química. 105, 4 (2009), 300–304.
[13] Perez García, H. 2012. Evaluación de los Museos Científicos del entorno de la Región de Murcia por su utilidad en la Educación Secundaria. Universidad de Murcia.
[14] Popat, S. and Starkey, L. 2019. Learning to code or coding to learn? A systematic review. Computers & Education. 128, (Jan. 2019), 365–376.
[15] Al Subhi, N.N. and Bell, D. 2018. Hybrid User Experience (UX) Modelling in a Heritage Setting. 2018 Winter Simulation Conference (2018), 1298–1309.
[16] Wenglinsky, H. 2005. Using technology wisely: The keys to success in schools. Teachers College Press.