Online Learning Facilities to Support Coding and Robotics Courses for Youth

https://doi.org/10.3991/ijep.v8i3.8044

Eleftheria Demertzi(✉), Nikitas Voukelatos,
Yannis Papagerasimou, Athanasios Drigas
N.C.S.R. "Demokritos", Athens, Greece
demertzi.e@gmail.com

Abstract—Nowadays programming and computational skills are of great importance in working and social life. Knowing how to code is empowering. It allows to understand the digital world we live in and to shape it. Basic coding skills are essential for accessing the jobs of today and tomorrow and for achieving a better skills-match between education and the labor market. This paper presents a European Project entitled "Coding and Youth: An innovative program in the digital era" (Code@Youth), which attempted to utilize constructively the long period of summer vacations in European countries in order to introduce students to the world of programming and robotics by offering hybrid learning activities in parallel with on-line learning facilities. The main purpose of this program was to plan, implement and evaluate these activities for young people, through the on-line platform, in order to lead to the acquisition, recognition and validation of computational skills obtained through non-formal learning.

Keywords—e-platform, e-skills, on-line learning facilities, hybrid learning activities, coding, computing, programming, robotics, non-formal education, summer school

1 Introduction

Programming and computational thinking skills are becoming more and more important in our society and working life. The European Union estimates that by 2020 up to 750,000 new Information and Communication Technology (ICT) professionals will be needed for the European job market [1], whereas computer science skills are increasingly required in many different fields, not only in ICT jobs. Today, a plethora of educational organizations are refocusing their ICT curricula on developing students' computer programming and coding skills, and introducing the topic in national, regional or school curricula.

Today's generation of students are extremely different learners from past generations [2]. These students have been called Digital Natives [3], the Net Generation [4], the Millennial Generation [5] and the Generation M [6] these individuals are said to have been 'born digital' [7]. In fact, students are so drastically different from past generations that most educational systems are not keeping pace [3,4,8].

Researchers highlight that one of the major concerns for modern education is to keep up with today's technologically confident youth [9]. So far, formal education has been using ICT to focus purely on computer literacy, teaching students how to word-process, how to work a spreadsheet and how to surf the Internet. These skills are rather important but on their own are failing to prepare students for their future studies and careers. The new socio-technical reality of the 21st century requires students to possess skills and abilities related to the use of technological tools [10], but also to be "digitally literate", meaning to have the skills, understandings and practices that are essential in "navigating the ever-changing digital landscape" [10].

To be digitally literate does not only concern being competent in school-based tasks, but also to be able to participate effectively in the new digital world [10]. A lack of digital literacy leads to implications on one's potential to be a competent student, an empowered employee or an engaged citizen [10]. As a result, there is nowadays an ever growing need to teach the next generation computer science, ICT and digital literacy, thus going beyond the skills of how to use a computer, to teach young people how a computer works and how to make it work for them by coding and creating programs.

Through programming students are exposed to computational thinking [11]. Computational thinking is "a way of solving problems, designing systems and understanding human behavior by drawing on the concepts fundamental to computer science" [12]. Through computational thinking, which is a problem-solving process, students are able to use computers and other tools to solve problems [13]. This is possible "by logically organizing and analyzing data, by representing data through abstractions such as models and simulations, by automating solutions through algorithmic thinking (a series of ordered steps), by identifying, analyzing, and implementing possible solutions with the goal of achieving the most efficient and effective combination of steps and resources, and by generalizing and transferring this problem-solving process to a wide variety of problems" [13].

The way work is done today has changed radically after the application of computer technology to every field of study [13]. The human mind remains the most advanced problem-solving tool but nowadays we have the ability to extend its strengths by using digital tools [13]. Computational thinking is not just for programmers anymore but is gradually becoming a ''universally applicable attitude and skill set" everyone would be eager to learn and use [12]. It also affects many different aspects of competencies needed in the 21st century such as creativity, critical thinking and problem-solving [14]. This is why it is important for everyone to learn to program as a part of a liberal education [15].

By programming and using computational thinking skills, students use computer science concepts, such as abstraction, debugging, remixing and iteration, in order to solve problems [16,17,18]. Programming students have talked about learning how to code as something unique. They claim programming provides a special way to think, different from any other subjects that they have studied [19]. The aforementioned dictate the need of exposing students to such concepts as early as possible in their school years in order to obtain these essential skills and learn how and when to use them [20].

Programming is not an easy subject to be studied or taught. In order to program, a student must understand correctly abstract and complicated concepts [21]. As a result many students experience learning problems [21]. To avoid such problems teachers must carefully design learning approaches, activities and material [21]. This will lead to robust knowledge and skill construction.

A domain that promotes computational thinking is robotics [22]. "Robotics represents a powerful, engaging tool for youth learning because students can touch and directly manipulate the robots, resulting in hands-on, minds-on, self-directed learning" [23]. Students learn experientially by involving in robotics activities. This is similar to problem-based learning in which students learn through authentic experiences by solving problems [24]. In a robotics course, students program robots with embedded code in order to achieve the desired functionality. They need to think about "how the robotic agent will interact within its world" [22] based on the instructions they give it.

Robotics is truly motivating for children [25]. Through it young programmers "approach technology both amusingly and intuitively, while discovering the underlying science principles" [25]. Students enjoy robotics courses because they combine theory with practical challenges [26]. Robotics offers special educational benefits, because "it is multi-disciplinary and involves a synthesis of many technical topics, including algebra and trigonometry, design and innovation, electronics and programming, forces and laws of motion, and materials and physical processes" [27].

Research shows that robotics can increase learning in specific Science Technology Engineering and Mathematics (STEM) concepts [28,29,30]. Internationally there is an interest in investing in STEM educational programs as countries are trying to compete in the global market place and in increasing the number of youth pursuing STEM careers [31]. As youth unemployment is a major issue all over Europe, mastering these skills is becoming critical to fill the huge gap created by the digital revolution.

Taking into consideration the above and the long non-working time of formal schooling in European countries, the Code@Youth program aimed to introduce the students to the worlds of Programming and Robotics through non-formal education. Non-formal education concerns educational activities that take place outside of school. Digital literacy can also be cultivated through non-formal venues of learning and development [10]. "What was once perceived as 'the school's responsibility' to prepare students to be digitally literate citizens is now the acknowledged responsibility of all learning spaces, formal and informal in order to ensure both preparation and continuous updating of digital literacy skills, understandings and practices for everyone from toddlers to seniors" [10].

2 The Code@Youth Project

Summertime has been seen as an opportunity for the Code@Youth project which aimed to design, implement and evaluate a non-formal set of hybrid learning activities of coding and robotics for young children in parallel with on-line learning facilities. This project intended not only to open up new routes into teaching coding and programming to teenagers, but also to open up their career options by introducing them to digital fields and improving their soft and digital skills.

The project involved a large number of students and instructors from four participating countries (Cyprus, Germany, Greece and Italy). The direct target group was students aged 13 to 16 years, with basic computer knowledge and skills. No programming or robotics knowledge was required. The indirect target group was instructors and trainers, or youth workers as they are called within the project, whose profiles were upgraded and strengthened through the participation in this non-formal summer program.

The aims of the project, apart from utilizing summer time constructively by teaching young students how to code, were: to provide opportunities for them to expand their spectrum of career prospects and to develop their soft skills by getting to know the various academic paths and careers related to the computing field; to build a robust e-platform as a support, communication and validation tool throughout the project; to support the changing role of young teachers in order for them to provide high-quality services based on building competences, raising quality standards, enriching expertise and reinforcing links between policy, research and practice; and to create synergies between various stakeholders such as Youth Centers, Research Centers, NGOs, ICT Providers, Institutions, Enterprises etc.

Based on the current scene in partner countries and the experiences from everyday practice, as well as from participation in other related projects, a great urge and interest emerged for the promotion of digital literacy in non-formal settings and improvement of career orientation towards computer related jobs. This initiative was expected to support the efforts towards bridging the skills and technology gap between education and the labor market. At the same time, it would promote the digital agenda of the participating countries, thus encouraging and opening up the prospects of young people to study computer related subjects, while improving the learning outcomes of young people.

The main aspect of the project was the actual preparation and realization of the hybrid learning activities in parallel with the Code@Youth e-platform facilities.

2.1 Planning the Program

As mentioned above, partners identified the need to introduce coding at a young age through an innovative program that would be challenging, creative and rewarding, while at the same time moving away from strict subject boundaries and formal curriculum restrictions, and providing first-hand experience with the professional and academic ICT community through visits and enlightening discussions. It was decided that the Code@Youth program would be based on a set competence framework, using existing successful tools and resources to be reviewed, accompanied by a powerful e-portal. The latter could analyze young peoples' progress, validate their coding skills, offer them professional training and sustainable learning.

A preliminary gap and needs analysis was conducted by all partners on their national policies and practices, about the needs of the labor market in relation to the skills of young people in the digital era. After the survey was conducted the final proposal of the project was composed taking in consideration the needs of all participating countries.

It was decided that the Code@Youth project would have three stages. The first stage was constituted of the preparation of the hybrid learning activities. This stage included the research and analysis mentioned above. The second stage concerned designing and developing these activities based on the needs that the first stage indicated. In this stage, all aspects regarding the organization, management, planning and implementation of the program, and specifically the summer school, were defined and designed. The final stage of the project was about the actual realization of the hybrid learning activities in parallel with the e-platform and their evaluation.

2.2 Code@Youth e-Platform

One of the main objectives of the project was to design and implement a dynamic and interactive e-tool that would support and facilitate the purposes of the project and would function as an Open Learning Environment within the project and in the future. This platform would offer on-line services that would support the learning activities of the program. The platform was designed as an e-class, in order to organize and make course material available and to facilitate communication between participants.

The platform was implemented with a set of options that make it an important e-tool for Coding and Robotics and in general. It was built with interactivity options in order to offer learning opportunities, like material, courses etc., through the e-Code@Youth Academy. The platform was decided to be continuously developed through the e-Code@Youth Resources Data Bank with useful e-tools and resources to support youth workers and students. Finally, it was decided that it will be connected with the e-Code@Youth Community for mentoring, peer and expert support, guidance and exchange of good practices.

This e-tool was expected to stimulate and empower even further the professionals and the students to be involved in the implementation of the project. It was meant to provide opportunities for the strengthening of the digital competences of youth workers and to ensure continuous development since open access and interconnectivity would be available. At the same time it enabled opportunities for open, free and self-regulated learning. The provision of all products in an open and accessible form provides added value to the project and ensures its sustainability and exploitation.

The Code@Youth e-platform was designed, developed and uploaded as a dynamic and interactive tool that would provide a set of activities related to the project's purposes. These activities were designed and implemented in order to facilitate and support the educational process for each category of users accordingly. Users can either be youth workers or students.

Every user was asked to register online and create their own profile in the platform exclusively designed for the Code@Youth project. In their profile users filled some personal information. After that, and throughout the program, depending on whether they are students or youth workers, they were asked to provide some information about their knowledge and experience and what they achieved through it. As a result their profiles are like 'portfolios' - working digital CVs for coding.

After the creation of their profile users could log in using their credentials and could ask for enrollment in specific courses. As mentioned above, the available courses within this project were the programming course named “Programming Basics” and the robotics course named “Robotics and Coding”. Once a user was enrolled in a course he or she could have access to the course’s educational material. Depending on the course the user could find basic information about it and have access to the syllabus or any other resources (glossaries, notes, videos, links, etc.) available for that specific course.

One of the e-platform’s main goals was to motivate the students to get involved as much as possible in the hybrid learning activities and study the educational material. For this purpose, the syllabus was decided to be divided in sections. Each section had a specific educational object and was subdivided in chapters. For each of these sections an electronic book was created. Having an e-book seemed to motivate the students greatly to study each course’s material, especially due to the fact that it was divided into small chapters and it was enriched with audiovisual material.

Another goal of the platform was to make assessment methods available to users. Bearing in mind that the learning in this program was non-formal, the assessment was not strict, and its purpose was mostly to help students to monitor and validate their progress and knowledge and to motivate them further. Students could easily monitor their progress in studying for one specific course through a checklist that was created. Through this checklist students and their teachers could view what educational material they studied and to what extent.

Another assessment method that seemed to inspire students greatly was the use of the Open Badges System and its incorporation into the platform. Open Badges is an innovative system used in the USA and many EU countries for the validation and recognition of learning. It is a visual verified evidence of achievement. It has visual part (image), an issuer, a date of issue, a description of the badge, criteria in order to be earned, evidence of what the badge owner is claiming, specific competence framework and tags. Open Badges were created on-line through the use of the platform.

The use of Open Badges motivated the students to participate in all learning activities by giving them a feeling of program gamification. Students learned in a positive way by trying to complete the criteria for earning each Badge, which was issued to them by their instructors, through the platform, and after taking into consideration their progress in the platform (checklists) and in the classroom. In that way, both students and teachers knew how to assess and evaluate the progress that had been made. Therefore e-Badges System functioned as a tool for transparency, validation and recognition of learning outcomes.

Through the platform students could record their progress in relation to their coding skills, the results of the quests - challenges taken to earn awards, the levels achieved, the Badges awarded. As a result their profile is like a 'portfolio’, a CV for future use. Through their e-profiles young people were able to show their achievements, motivate others to participate, whereas youth workers were able to use this method in order to motivate their students to participate in learning activities and help them towards obtaining desired benefits. Finally, in their profiles users could upload any other evidence of achievements accomplished.

Another important use of the Code@Youth e-platform was that it provided various communication methods for users of each course. There was synchronous communication through chats and asynchronous communication through forums and various announcements. In that way the members of each classroom could communicate among them as well as with other participants of the program from other countries. Specifically for youth workers there was a special forum where they had the chance to exchange views and good practices for the benefit of the learning progresses.

Finally, through the platform users and any other interested parties could have access to the program's social media pages. In this way they could get informed about the program, learn about its goals and aims and follow its implementation, step by step, through the various posts made by its participants. This aimed at the transparency of the program's actions and also at the dissemination of the program and its results.

2.3 Hybrid Learning Activities

As mentioned above and beyond the online learning facilities provided, one of the main purposes of the Code@Youth project was the actual realization of the hybrid learning of coding and robotics.

Preparation of Hybrid Learning Activities – After the first stage of the project, which included the research and analysis, was completed and the proposal was finalized, the planning and organization of hybrid learning activities began. These hybrid learning activities that would be in parallel with the on-line learning facilities mentioned above would have the form of summer school lessons. Decisions had to be made concerning the syllabus and the educational material that was going to be taught. The latter were decided to be kept simple, bearing in mind that this was an informal summer school, that would last two weeks, and that the participants didn't have any previous programming or robotics knowledge. The educational material was decided to be divided in two levels. The first was the Programming Level and the second was the Robotics Level. Each partner could decide when it was best for them to hold the summer school and invite the participants to apply.

Realization of Hybrid Learning Activities – The first part of the training focused on the acquisition of the Basic Coding Skills. Students were introduced to Basic Coding Concepts (problem, data, information, variable, constant, data type, expression, algorithm, flowchart, table) and Basic Structures (Sequential Structure, Iteration Structure) in Visual Basic Coding Language. Guided by the instructors, students were able to work on basic and more complicated problems and exercises in order to familiarize with coding. The second part of the training concerned the introduction of the Students to the world of Robotics. Using the basic Coding Skills obtained in the first part students had the opportunity to learn how to program an Edison Robot using a graphic interface.

Over 40 hours of training were allocated. Particularly, the training had the following structure: 8 hours of Basic Coding Concepts (problem, data, information, variable, constant, data type, expression, algorithm, flowchart, table, structures), 12 hours of Visual Basic Coding Language (exercises, algorithms and programs using Visual Basic), 4 hours of Basic Robotics Skills (introduction to Robotics, getting to know Edison Robot and Basic Edison Programs using Edison Barcodes), and 20 hours of Robotics (basic and more advanced Edison Controls and Programs). Each partner modified the hours spent on the Coding and Robotics training based on the needs of the students as identified in the results of the preliminary research, as well as on the real needs of the actual participants.

Some of the main issues encountered during the Summer School were firstly the different level of understanding (mainly of concepts) among the participants due to the age difference, the lack of interest in the more theoretical concepts on the Coding lessons and the anticipation to move to more interesting parts such as the Visual Basic exercises and the programming of Edison Robots. Another issue encountered was that training time was limited considering that students didn't have homework or studying time outside the classroom. As a result there wasn't always enough time for practicing on exercises and problem solving. For that reason instructors added some extra hours for the students that wanted to practice more on exercises and also tried to motivate students to exercise even after the end of the Summer School by having extra activities available on the e-platform.

Evaluation of Hybrid Learning Activities – Bearing in mind that this was an informal summer program, no official evaluation of the training took place. Nevertheless, it was important to have feedback on the process, so the evaluation was done informally within the classroom. On the last day of the training, instructors asked students to comment on the course. Through discussion, students were asked about the course content, the course delivery, the communication and the support by the instructors and also the e-tool available. The feedback was positive as students stated that they felt they were using their free summer time constructively by learning things not taught at school and obtaining digital skills. Everyone involved had positive things to say about the course content and delivery as well as the communication and support, except for the fact that the vast majority complained about the limited time of the program. Students underlined that this program was really different compared to traditional school lessons because they had the chance to learn by having active participation in exercises and actions that were entertaining.

3 Conclusions

There is nowadays an ever growing need to teach computer science, ICT and digital literacy to the next generation. In today's digital era, students need to possess knowledge and skills to use technological tools but also to be digitally literate, in a way that will allow them to successfully participate in the digital landscape that is constantly changing. We have to go beyond the skills of how to use a computer. We need to teach youth how a computer works and how to make it work for them by coding and creating programs.

Computational thinking provides a set of skills that everyone should possess and use. It is a new way of thinking and students should be exposed to it as early as possible. Robotics is a field that promotes computational thinking. Through it students learn experientially in a self-directed way that combines theory and practical challenges. Robotics motivates students highly, is multi-disciplinary and increases STEM skills. Globally and in order to compete in the labor market there is a growing interest in STEM education and STEM studies and careers.

The idea for the Code@Youth project was born in this context and framework and aimed at developing, implementing and evaluating a comprehensive challenging summer program that would focus on introducing young people to the digital world, by using long summer vacations constructively. This will introduce youths to the spectrum of fields they can follow to study and later on work, within the digital area, and should supply them with the e-skills needed to bridge the technology gap between education and the world of work.

The results of the Code@Youth project were very gratifying and fulfilling for everyone involved. Students acquired basic coding and robotics skills that introduced them to digital fields, while at the same time they had the opportunity to enjoy themselves with engaging experiential activities with the robots.

Through the Code@Youth project a new on-line platform was created. The platform has in-built functions of interconnectivity, communication and provision of support through the e-Academy, where experts and stakeholders can register to offer guidance and support and an e-databank with useful e-tools, reports, good practices etc. for review and the e-community where young people can share their experiences and learn from each other.

The project managed to introduce quality standards, evidence-based data and mechanisms for the validation of the acquired coding competences for young people through the system of Open Badges. This was an added value to the project as the coding skills of the students are visible, transparent and accessible through the e-platform.

According to the participants themselves, the Code@Youth project was “a unique experience”, as it has given them the opportunity to learn experientially, in a really different way and context to the ones at traditional schools. All of them claimed that through this program they now have a different view on the digital fields and their future studies and careers.

Students showed real interest in continuing this coding and robotics hybrid learning activities and hope for a new series of training sessions that would cover more topics in the future. The continuation of such multi-disciplinary learning activities will give youth the opportunity to become active members of the new, ever-changing digital landscape. Policy and decision makers should take the aforementioned facts into consideration and continue to develop policies and strategies that will ensure the promotion and financing of such activities.

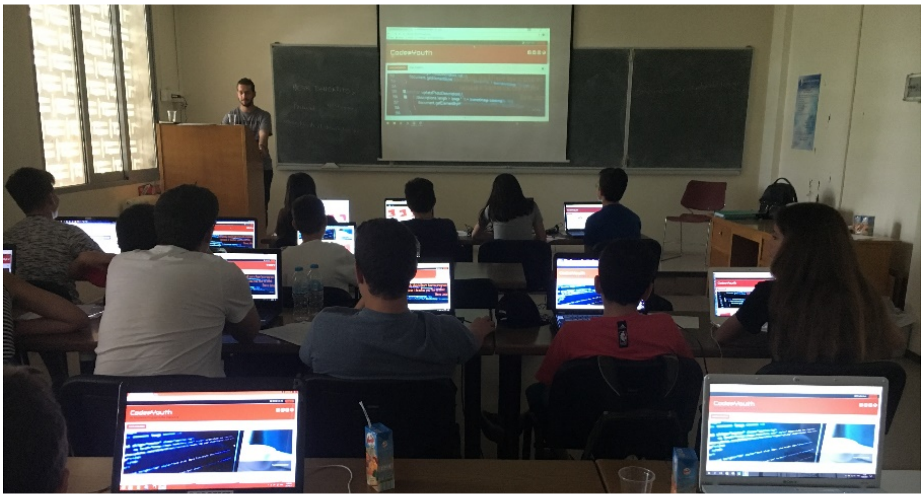

Fig. 1. Students using the e-platform.

4 Acknowledgement

This paper is supported by the EU Project "Coding and Youth: An innovative programme in the digital era" (Code@Youth), which is an ongoing Erasmus+ Strategic Partnerships for Youth Project (Project number: 2016-2- CY02-KA205-000819).

5 References

[1] European Commission (2011). Europe 2020: Flagship Initiative Innovation Union

[2] Nasah, A., DaCosta, B., Kinsell, C., & Seok, S. (2010). The digital literacy debate: an investigation of digital propensity and information and communication technology. *Educational Technology Research and Development*, *58*(5), 531-555. https://doi.org/10.1007/s11423-010-9151-8

[3] Prensky, M. (2001). Digital natives, digital immigrants part 1. *On the horizon*, *9*(5), 1-6. https://doi.org/10.1108/10748120110424816

[4] Tapscott, D. (1998). *Growing up digital: The rise of the net generation* (Vol. 352). New York: McGraw-Hill.

[5] Howe, N. (2000). Millennials rising: the next great generation./Neil Howe, William Strauss.

[6] Roberts, D. F., Foehr, U. G., & Rideout, V. (2011). Generation M: Media in the Lives of 8–18 Year-Olds. Menlo Park, CA: Henry J. Kaiser Family Foundation; 2005.

[7] Palfrey, J. G., & Gasser, U. (2011). Born digital: Understanding the first generation of digital natives. ReadHowYouWant.com.

[8] Oblinger, D. G., & Oblinger, J. L. (2005). Educating the net generation. Boulder, CO: Educause.

[9] Bennett, S., Maton, K., & Kervin, L. (2008). The 'digital natives' debate: A critical review of the evidence. *British journal of educational technology*, *39*(5), 775-786 https://doi.org/10.1111/j.1467-8535.2007.00793.x

[10] Meyers, E. M., Erickson, I., & Small, R. V. (2013). Digital literacy and informal learning environments: an introduction. *Learning, media and technology*, *38*(4), 355-367. https://doi.org/10.1080/17439884.2013.783597

[11] Lye, S. Y., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12?. *Computers in Human Behavior*, *41*, 51-61. https://doi.org/10.1016/j.chb.2014.09.012

[12] Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, *49*(3), 33-35. https://doi.org/10.1145/1118178.1118215

[13] Barr, D., Harrison, J., & Conery, L. (2011). Computational thinking: A digital age skill for everyone. *Learning & Leading with Technology*, *38*(6), 20-23.

[14] Ananiadou, K., & Claro, M. (2009). 21st century skills and competences for new millennium learners in OECD countries.

[15] Perlis, A. (1962). The computer in the university. *Computers and the World of the Future*, 180-219..

[16] Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 annual meeting of the American Educational Research Association, Vancouver, Canada* (pp. 1-25)

[17] Ioannidou, A., Bennett, V., Repenning, A., Koh, K. H., & Basawapatna, A. (2011).Computational thinking pattern. In Annual American Educational Research Association meeting. New Orleans, Louisiana, United States.

[18] Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical transactions of the royal society of London A: mathematical, physical and engineering sciences*, *366*(1881), 3717-3725. https://doi.org/10.1098/rsta.2008.0118

[19] Eckerdal, A., Thuné, M., & Berglund, A. (2005). What does it take to learn'programming thinking'?. In *Proceedings of the first international workshop on Computing education research* (pp. 135-142). ACM.

[20] Yadav, A., Zhou, N., Mayfield, C., Hambrusch, S., & Korb, J. T. (2011). Introducing computational thinking in education courses. In *Proceedings of the 42nd ACM technical symposium on Computer science education* (pp. 465-470). ACM. https://doi.org/10.1145/1953163.1953297

[21] Lahtinen, E., Ala-Mutka, K., & Järvinen, H. M. (2005). A study of the difficulties of novice programmers. In *Acm Sigcse Bulletin* (Vol. 37, No. 3, pp. 14-18). ACM. https://doi.org/10.1145/1151954.1067453

[22] Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., & Werner, L. (2011). Computational thinking for youth in practice. *Acm Inroads*, *2*(1), 32-37. https://doi.org/10.1145/1929887.1929902

[23] Nugent, G., Barker, B., Grandgenett, N., & Welch, G. (2016). Robotics camps, clubs, and competitions: Results from a US robotics project. *Robotics and Autonomous Systems*, *75*, 686-691. https://doi.org/10.1016/j.robot.2015.07.011

[24] Barrows, H. S. (1996). Problem-based learning in medicine and beyond: A brief overview. *New directions for teaching and learning*, *1996*(68), 3-12. https://doi.org/10.1002/tl.37219966804

[25] Ruiz-del-Solar, J., & Avilés, R. (2004). Robotics courses for children as a motivation tool: the Chilean experience. *IEEE Transactions on Education*, *47*(4), 474-480. https://doi.org/10.1109/TE.2004.825063

[26] Rawat, K. S., & Massiha, G. H. (2004). A hands-on laboratory based approach to undergraduate robotics education. In *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on* (Vol. 2, pp. 1370-1374). IEEE. https://doi.org/10.1109/ROBOT.2004.1308015

[27] Johnson, J. (2002) Children, robotics, and education. In *Proceedings of the 7th International Symposium on Artificial Life and Robotics (AROB-7)*, pp. 491–496.

[28] Barker, B. S., & Ansorge, J. (2007). Robotics as means to increase achievement scores in an informal learning environment. *Journal of Research on Technology in Education*, *39*(3), 229-243. https://doi.org/10.1080/15391523.2007.10782481
[29] Nugent, G., Barker, B., Grandgenett, N., & Adamchuk, V. I. (2010). Impact of robotics and geospatial technology interventions on youth STEM learning and attitudes. *Journal of Research on Technology in Education*, *42*(4), 391-408. https://doi.org/10.1080/15391523.2010.10782557
[30] Williams, D. C., Ma, Y., Prejean, L., Ford, M. J., & Lai, G. (2007). Acquisition of physics content knowledge and scientific inquiry skills in a robotics summer camp. *Journal of Research on Technology in Education*, *40*(2), 201-216. https://doi.org/10.1080/15391523.2007.10782505
[31] Langen, A. V., & Dekkers, H. (2005). Cross-national differences in participating in tertiary science, technology, engineering and mathematics education. *Comparative Education*, *41*(3), 329-350 https://doi.org/10.1080/03050060500211708

6 Authors

Eleftheria Demertzi holds a Bachelor degree in Informatics (2016). She is a Research associate at N.C.S.R. "Demokritos" on national and European ICT projects. She has participated in the development, research, study and design of multimedia and network applications (e-services) such as e-learning, e-culture, e-inclusion, etc. (email: demertzi.e@gmail.com).

Nikitas Voukelatos holds a Bachelor degree in Informatics (2017). He is a Research associate at N.C.S.R. "Demokritos" on national and European ICT projects. He has participated in the development, research, study and design of multimedia and network applications (e-services) such as e-learning, e-culture, e-inclusion, etc. (e-mail: nikitas.vouk@gmail.com).

Yannis Papagerasimou holds a Masters degree in Data Communications Systems (2003) from Brunel University where he also graduated with a Bachelor in Electronic and Computer Engineering in 2001. Since 2005 he has been a Research associate at N.C.S.R. "Demokritos" on national and European ICT projects. He has participated in the development, research, study and design of multimedia and network applications (e-services) such as e-learning, e-culture, e-inclusion, etc. He has also published more than 10 scientific articles to conferences and scientific journals.

Athanasios Drigas is a Director of Research at IIT - N.C.S.R. Demokritos. He is the Coordinator of Telecoms Lab and founder of Net Media Lab since 1996. From 1990 to 1999 he was the Operational manager of the Greek Academic network. He has been the Coordinator of Several International Projects, in the fields of ICTs, and e-services (e-learning, e-psychology, e-government, e-inclusion, e-culture etc). He has published more than 280 articles, 7 books, 25 educational CD-ROMs and several patents. He has been a member of several International committees for the design and coordination of Network and ICT activities. (e-mail: dr@iit.demokritos.gr).

Article submitted 30 November 2017. Resubmitted 05 February and 01 March 2018. Final acceptance 02 March 2018. Final version published as submitted by the authors.