Changing Students' Attitude towards STEM by Educational Robotics & Digital Games Programming.

Batzogiannis Ilias[1], Hatzikraniotis Evripidis[2]
[1] P technologies
[2] Associate Professor, Physics Department, AUTH
(Greece)
[1]iliasbatzo@yahoo.gr, [2]evris@physics.auth.gr

Abstract

This paper suggests the introduction of the educational robotics and the design of digital games in secondary school education aiming at changing students' attitude towards STEM. The idea is to engage students in the educational process by constructing and programming. The theoretical background of the methodology in both methods is learning by inquiry. Two methods for students' engagement were studied, educational robotics and digital games programming. The tool being used in the first method is Lego Mindstorms Ev3 that offers a great variety of constructing and programming potentials. In the second method the tool that we use is Microsoft Kodu Game Lab Software which can be a great introduction to programming. Both methods are implemented in 12-13 year old students attending the first class of Greek high school for ten hours. In order to test the results of the methods implemented, of how does students' attitude towards STEM change, pre and post (Likert scaled) tests are answered by the students. Both methods were found influential for students to develop a more positive attitude towards STEM, with digital games programming to perform slightly better than robotics.

Keywords: educational robotics, attitude towards science, teaching science

Introduction

It has been observed in recent years that students are drawn away from Science Technology Education and Mechanics (STEM). This is due to the difficulty of the subject, the non-connection of what they learn to real life and the fact that students do not "learn" STEM. The only thing that they do learn is solving "some equations" [1].

The investigation of students' attitudes towards studying science has been a substantive feature of the work of the science education research community for the past 50 years. The increasing attention to the topic is driven by recognition that all is not well with school science and far too many pupils are alienated by a discipline that has increasing significance in contemporary life, both at a personal and a societal level [2]. As Osborne states in his review article, «while it would be difficult to transform the nature of science offered in most curricula, at least in the short term, a better understanding of the attributes of science classroom activities that enhance 'task value' might make a significant contribution to how the quality of students' experience might be improved». Thus, innovative learning activities, such as the employment of educational robotics, or the designing and programming of a digital game may provide with the enhanced 'task value' and therefore influence the students' attitudes towards science.

Educational robotics is the tool that we will use in our first attempt to alter this situation. We believe that as students are designing, building and programming these robots, they will be motivated to learn STEM needed to achieve their goal [3] [4].

Nowadays, the teacher has the means to associate the theory of STEM with real life through educational robotics [5] [6] [7]. The students realize that they really use what they learn. The tool which will help us to achieve this goal is the Lego Mindstorms EV3.The versatility of the hardware and software allow a wide variety of possibilities in what students can build and program. Lego Mindstorms have been used by teachers all over the world in teaching STEM [8].

In our second research, the tool being used is Kodu Game Lab software. By using this program students are able to design their own digital game. They can decide how the scenery will look like, the gameplay, the heroes, the goal e.tc. By creating their own game students realize that they need scientific knowledge such as mass, velocity e.tc in order to create their own world and they express and explore fundamental computer science concepts [9],[10]. Kodu has also been used in many researches showing some promising results in improving the motivation for learning programming concepts [11]

The main educational method being used in both researches is the inquiry method because we believe that children may reach higher levels of understanding science when they perform structured investigations [12] [13]. Other methods being used are the constructional method and constructivism. In this work, we examine how students' attitudes towards STEM are influenced by innovative learning applications, such as educational robotics and digital games programming.

Description of the Sequence

In this research, we are trying to change students' negative attitude towards STEM by using two approaches, namely (first) educational robotics and (second) digital games programming. Both studies implement a ten-hour program into two different groups in the first grade of secondary school (12 year old students) in Platon School of Katerini, Greece. 27 students were participated in the "robotics" group and 30 in the "game programming". Questionnaires had been given to them to examine their attitude towards STEM, before (pre-test) and after the intervention (post-test).

Robotics

In this research, Lego Mindstorms are used as "friendlier" towards the students. Lego bricks, sensors and an "EV3 brick" (microcomputer) are used to construct the robot. In our research, the robot is constructed by the students easily by following some instructions given (picture 1). The robot is a rover.

Students were divided into groups of five or four with clear, but different roles each time (coordinator, manufacturer, developer, etc.). Care was taken so that each groups consisted of students with different grades, different attitudes towards STEM and different genders in the same team.

Picture 1: Rover

Picture 2: Lego software

The ten-hour intervention consisted of "missions". The "missions" guide the students to set the rover to fulfill its purpose. In order to do so, the rover must be capable of avoiding objects (using its sensors). During this procedure the students learn things about STEM (physics: velocity, force, friction, reflection of light e.tc, learn things about math: unit's conversion, angles e.tc, acquire scientific skills: carry out measurements, draw graphs, problem solving and also acquire programming skills and computational thinking). These "missions" were assigned to each team through students' worksheets. Through this procedure they come across some problems that they have to overcome on their own. Of course these worksheets include some instructions given by us but as the program is progressing, less instruction is given. This is our way of making them inquire and not just follow instructions.

The programming of the robot is carried out by Lego Mindstorms software. The software is easy for the children to understand and use as it works with pictures (picture 2) [14].

Digital games programming

In this research we are using the Kodu Game Lab software. The students work in groups of two. We start by having a discussion about how the computer "thinks" and how we should "talk" to it in order to understand us. The programming is really easy and based on "when – do" (picture 3). Kodu includes a novel graphical programming environment based on a concurrent rule system without any code-writing involved [15]. We start with small tasks and as the implementation moves on the tasks get bigger and help given gets less and less (inquiry method). Similar to robotics, in this procedure the students also come across with concepts and practices about STEM (as described before).

Picture 3: Programming Kodu

The great thing with this implementation is that students try to make different games. Games that sometimes get really complicated and difficult to win. They are motivated as every group play they other's group game and there comes a competition of who created the best game. The graphics are not something great (picture 4) but this is not the point. The point is that they play with something that they created.

Picture 4: Designing the game

Description of the method

In order to check if our method had any impact on students' attitude toward STEM, we constructed a questionnaire on a Likert scale, which was given to students before the program (pretest) and at the end of it (posttest). The scale was:

Never/ Strongly disagree	Rarely/ Disagree	Sometimes/ Undecided	Usually/ Agree	Always/ Strongly agree

The questionnaire consists of 21 questions categorized in 5 categories which are:

1. 7 questions about themselves (gender, age, grade in physics, etc.)
2. 5 questions checking if the students are interested in physics and math (Q1: I observe natural phenomena and try to learn why they happen, Q2: I read science book and articles, Q3: I would like to have a job that has to do with science, Q9: I like science, Q10: I like math)
3. 4 questions checking how students feel or react during class (Q5: I care about science class, Q6: I participate during science class, Q7: I feel nervous during science class, Q8: I would like to experiment more during science class)
4. 3 questions checking if the students consider STEM important (Q4: it is important for someone to know science, Q11: I use what I learn in science class in everyday life, Q13: I don't need science)
5. 2 questions checking if the students think that STEM is easy (Q12: Science class is easy, Q14: Science is easy)

Results & Discussion

The statistical analysis was made using miniTab Statistical Software in two different groups, robotics (27) and games programming (30). We want to check what impact the sequence had on both groups. The scale used to assign the data was -2,-1, 0, 1 and 2.
In order to explore the change of attitude, we performed a Paired t-test for the difference post-pre for robotics and digital games programming. We will present here the results of one question of its category.

Category 2 (Q2: I read science book and articles)

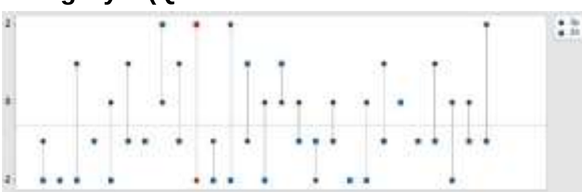

Figure 1: Robotics Q2 post (μ)-pre (π)

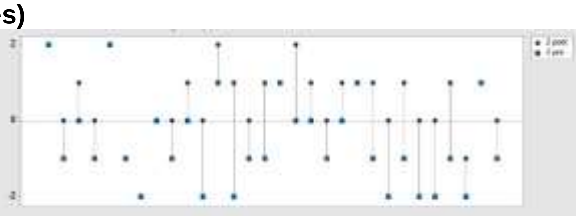

Figure 2: Games programming Q2 post (μ)-pre (π)

Figures 1 & 2 show the ranking for each student in Q2, for robotics and game programming. Squares represented the answers in the pre-test, while circles the post. The longer the lines connecting the two signs the greater the impact. As can be seen, both methods had a great impact in students' attitude. The statistical analysis for the robotics method gives ($P=0.015<0.05$) and for the games programming method ($P<0.01$).

Category 3 (Q5: I care about science class)

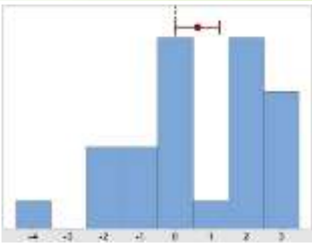

Figure 3: Robotics Q5 net change (post-pre)

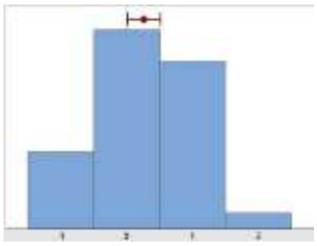

Figure 4: Games programming Q5 net change (post-pre)

Figure 3 & 4 represent the net change (post-pre), as bar-charts, for Robotics and Game-programming. In this question both methods also had a positive impact in students' attitude. The crossbar indicates the post mean while the dashed line indicates the pre mean. As it is seen the post mean has moved to more positive answers. The statistical analysis for the robotics method gives (P=0.049<0.05) and for the games programming method (P=0.036<0.05). The mean and standard deviation for the pre and post answers given for the Robotics is (0.74**,** 1.38) (1.37**,** 1.15) and for the Games Programming is (-0.1**,** 1.27) (0.17**,** 1.44).

Category 4 (Q11: I use what I learn in science class in everyday life)

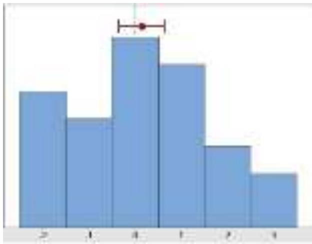

Figure 5: Robotics Q11 net change (post-pre)

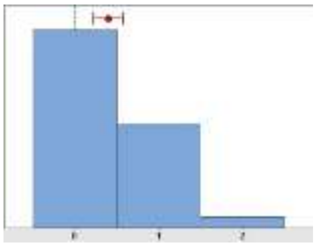

Figure 6: Games programming Q11 net change (post-pre)

Figure 5 & 6 represent the net change (post-pre), as bar-charts, for Robotics and Game-programming. In this question the results are a little different. We can say that the robotics method didn't have a statistically measured impact (the dashed line is within the crossbars in Fig 5) but on the other hand the game programming method had a large impact (the crossbar has moved to more positive answers). The statistical analysis for the robotics method gives (P=0.307>0.05) and for the games programming method (P<0.01). The mean values for the pre-test for both types of intervention (0.44 & 0.33) is shifted towards more positive (0.59 & 0.73) while the distribution gets narrower (standard deviation for pre is (1.12 & 1.03) and for post is (0.93 & 0.83).

Category 5 (Q12: Science class is easy)

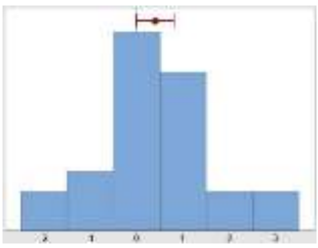

Figure 7: Robotics Q12 net change (post-pre)

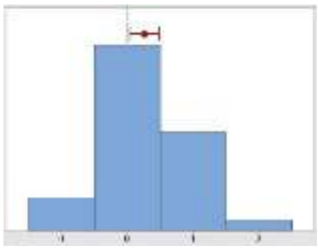

Figure 8: Games programming Q12 net change (post-pre)

Figure 7 & 8 represent the net change (post-pre), as bar-charts, for Robotics and Game-programming. We can say that both methods had a positive impact in the students with the games programming method showing more positive results (as it is seen from the crossbars). In this question the statistical analysis for the robotics method gives (P=0,05) and for the games programming method (P=0,022<0,05). The mean values for the Robotics is shifted from 0.00 (pre) to 0.41 (post) and for the Games

Programming from 0.43 to 0.70. As in the previous category, the standard deviation tends to decrease, from 1.11 to 0.97 for the Robotics and from 1.17 to 0.99 for the Games Programming.

Conclusion

As an overall conclusion, there is a positive change in the children's attitude toward STEM. According to their statements, they have started to observe natural phenomena and try to investigate the reason ("why") more often than before, they think that science class is easier than they thought e.tc.
Comparing the two methods, digital games programming seems to have a more positive influence in students' attitude towards STEM than robotics.
It seems like innovative learning applications, in this case educational robotics and digital games programming can influence the students' attitudes towards STEM. The research will continue with more students in order to verify our results.

References

[1] McDermott, L. C. (1991). What we teach and what is learned—Closing the gap. *American Journal of Physics,* 59 (4), 301-315
[2] Osborne, Jonathan, Simon, Shirley and Collins, Sue (2003). Attitudes towards science: a review of the literature and its implications, International Journal of Science Education, 25:9, 1049 – 1079
[3] Bers, M.U., Ponte, I., Juelich, C., Viera, A. & Schenker, J. (2002). Teachers as Designers: Integrating Robotics in Early Childhood Education. *Information Technology in Childhood Education Annual, 2002*(1), 123-145. AACE
[4] Frangou, S., Papanikolaou, K., Aravecchia, L., Montel, L., Ionita, S., Arlegui, J., Pina, A.. Menegatti, E., Moro, M., Fava, N., Monfalcon, S., Pagello, I. Representative examples of implementing educational robotics in school based on the constructivist approach Workshop Proceedings of SIMPAR 2008, pp. 54—65 (2008)
[5] Papert S. (1980), Mindstorms: Children, Computers, and Powerful Ideas, published by Basic Books
[6] Resnick M., Ocko, S. (1988) Lego /Logo Learning Through and About Design. In Papert S, Harel I. (ed.) Constructionism, pp.141-150. Ablex Publishing Corporation, US (1991)
[7] Williams, D.C., Ma, Y., Prejean, L., Ford, M.J. & Lai, G. (2008). 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
[8] Church, W., Ford, T., Perova, N., & Rogers, C. (2010). . In *AAAI Spring Symposium Series*. Retrieved from https://www.aaai.org/ocs/index.php/SSS/SSS10/paper/view/1062 (last visit 14/12/2014)
[9] Papastergiou, M. (2009). Digital game-based learning in high school computer science education: Impact on educational effectiveness and student motivation. Computers and Education, 52 (1), 1-12
[10] Stolee, K.T., and Fristoe, T. "Expressing computer science concepts through Kodu Game Lab." In Proceedings of the 42nd ACM Technical Symposium on Computer Science Education, (Dallas: Association for Computing Machinery, 2011): 99-104
[11] Fowler, A., Fristoe, T., and MacLaurin, M. "Kodu Game Lab: A programming environment." The Computer Games Journal, 1, 1 (2012):17-28
[12] Edward F. Redish and Richard N. Steinberg, Teaching Physics: Figuring Out What Works, Physics Today, Vol. 52 (January 1999), pp. 24-30
[13] Vosniadou, How do children learn, International Academy of Education, 2001
[14] http://www.lego.com/en-us/mindstorms/downloads/download-software (last visit 16/12/2014)
[15] MacLaurin, M. B. (2009). Kodu: end-user programming and design for games. Paper presented at the 4th International Conference on Foundations of Digital Games (FDG 09), Orlando, Florida