7th International Conference on Intercultural Education "Education, Health and ICT for a Transcultural World", EDUHEM 2016, 15-17 June 2016, Almeria, Spain

The influence of game-based programming education on the algorithmic thinking

Gabor Kiss* & Zuzanna Arki

J. Selye University, Bratislavská cesta 3322, Komarno, SK-94501, Slovakia

Abstract

Applied informatics engineering students of the undergraduate course Programming I get acquainted with the basic programming knowledge (structured programming) in C.

Our earlier personal experiences in Higher Education showed that the majority of the students can not acquire the algorithmic thinking by following the traditional teaching process and they do not have too much chance to pass it.

Our starting hypothesis was that the group where the students followed the game oriented exercises would achieve better results in the papers. After the semester we collected the paper results by groups and we tried to analyze whether this method was helpful or not for the students.

The traditional way to teach programming for applied informatics engineering students was not so successful than the game-based examples.

The students were more motivated to write game programs and more students could pass the test. It means by same learning material and test exercises the student who learned programming in game-based method could get more than one mark better paper results and pass the test in higher percent.

1. Introduction

The algorithmic thinking is important in all professions, especially in the engineering work by problem solving.

---

* Corresponding author. Tel.: +421353260/854; fax: +421353260/854.
E-mail address: kissga@ujs.sk

Applied informatics engineering students of the undergraduate course Programming I get acquainted with the basic programming knowledge (structured programming) in C.

The Slovakian students have just few chance to learn programming before finishing the secondary grammar school and make the final exam mainly we have to start with the basics (Kiss, 2012).

My earlier personal experiences in Higher Education showed that the majority of the students can not acquire the algorithmic thinking by following the traditional teaching process and they do not have too much chance to pass it. Unfortunately, I am not alone with this feeling. Three of the expert participants claimed that the difficulty in understanding the concept of programming and coding is because of the ineffective teaching strategies used during problem solving and coding (Ismail, Ngah, Umar, 2010). We can see all students have different levels of computer usage abilities by same Information Technology Education in other countries too (Isman, Celikli, 2009, Gastelú, 2013). It is the reason, that I think the creating of the algorithmic thinking by the undergraduate students need more time, or other teaching methods.

First of all we need to see how the human memory is working (Bloom, B.S.; Engelhart, M.D.; Furst, E.J.; Hill, W.H. and Krathwohl, D.R., 1956), the taxonomy of learning, teaching, assessing (Anderson, Krathwohl and Bloom, 2001) and the levels of learning to guide the students through the process of learning (Hoffmann, 2011). The performance of programming ability and thinking skills of students needs hard and concentrated work by teachers (Kurland, Pea, Clement, Mawby, 1986) and also subject Programming is important to attend the didactical methods of mathematic too (Ersoy, 2005).

Using playing games in teaching programming is productive (Esper, Foster, Griswold, 2013) and the kids can take the advantage of the different teaching methods. The situation does not change by students.

My experience shows the LEGO-Mindstorm is a very good tool for learning programming, because the students can construct a robot with different functions and write programs without syntax error (Kiss, 2010a). This tool is useful by teaching programming for girls too and the half of the economic information technology students are girls, who have more problems to learn programming (Kiss, 2010b) and they are not so motivated to learn programming than the computer science engineering students.

The subject Programming requires the logical and algorithmic thinking from the students and the teachers have to develop these skills in the students, but the teachers have to consider the motivation and the precognition of the students before starting to teach any topic.

Although I would be useful for the economic information technology students, I could not use LEGO-Mindstorm for teaching programming. I have to follow the formal of the programming subject and teach C as a first programming langue for these students.

I had an idea to write card games and dice games in my course.

My starting hypothesis was that the group where the students followed the game oriented exercises would achieve better results in the papers.

After the semester I collected the paper results by groups and I tried to analyze whether this method was helpful or not for the students.

2. Method

When the students could use the basic program elements like selection, iteration and can use the random generator in C after understanding the rules of the games, they could write a dice or a card games. The teacher can show the students how to use selection and iteration to build the ground for the game programs. For example the student play a dice game against the computer. The developed program generated randomly two (rolled) dice value, one for the student, one for the computer. We can choose a winner (who has higher value) by using selection. Another example can show how we can use the iteration combined with selection: the developed program generate randomly (rolled) dice values (more than one) for the two players (student and computer). The winner who has more rolled/generated "six" value. After this basic knowledge the students are able to combine these elements and write/develop complex game program with more rules.

We could not draw graphic for the game, we used just standard output opportunities to write the results after rolling the dices or taking a card and the computer was the second player with own decision progress. In this case the students could simulate a game against a computer. The students were very motivated to write different type of

games (black jack, dice poker and so on) and they could understand easier how we can use the different programming elements to write complex program. The students were more interested for these examples than the traditional programming examples. I supposed those who learned programming by following this teaching method would have better paper results later.

I have tested this method successfully earlier by economic information technology students in the undergraduate course Programming I. in Hungary, but in that case in different circumstances with two colleagues. My colleagues used the conventional teaching methods that they use with engineering students in teaching programming and I used my evolved way and we had the same time to teach the learning material (Kiss, 2013).

I had an excellent opportunity to use this method in other country and the person of the teacher is same that time in the two groups, so just the method is different the rest of circumstances are same.

Applied informatics engineering students of the undergraduate course Programming I. attended my lectures held at different times in two groups in first semester. The build of groups based on random, because the students' office separated the students in these two groups before. The full population of the students was 63 and I thought 34 students in this semester in two groups.

In the first group (group A) of lectures (20 students) I followed the traditional way with normal examples from real life while the courses for the second group (group B, 14 students) I made the course by using the game oriented teaching methods. The learning material was the same in the two groups, just the type of examples and exercises were different. Now I wanted to know the usefulness of all the game oriented examples and exercises I used.

3. Analysis of the paper results

Some mathematical analysis was needed to decide whether using these game oriented exercises were helpful or not in understanding the lectures and get better paper results.

3.1. The Number of Participants in the Tests in the Two Groups and the Values of Mean and Std. Deviation

According to the table (Table 1.) the mean of the results of papers in group B is higher. This group wrote the papers with a better result (six level of paper result is in Slovakia). It does not give enough information to state that the use of game-based programming education results in better written tests because this can happen accidentally, too. So, we needed more analyzing to keep the chance of accident low. I used the IBM SPSS Statistics v.20 by analyzing the paper results.

Table 1. Group statistics of the test results

Group	Number of participants	Mean	Std. Deviation
A	20	3.70	2.28
B	14	5.71	1.07

If we spend more time looking at this table, we can see 1.07 Std. Deviation of applied informatics engineering students who took part in the game-based programming education and the students who used the traditional way of education in higher value, but we still do not know if it is a coincidence.

3.2. Independent samples test

My null hypothesis was that the results of the paper written by the two groups of students would not differ significantly. Since we have two independent samples, we can use the independent sample test in SPSS to tell if the means of the paper of these groups differ or not (Table 2).

Table 2. Independent samples test of the test results

	Levene's test for Equality of variances	T-test for equality of means

	F	Sig.	t	df	Sig. (2-tailed)
Equal variances assumed	32.16	0.00	-3.13	32	0.00
Equal variances not assumed (Welch's t-test)			-3.51	28.98	0.00

An analysis of the results of the applied informatics engineering students showed, the variance of two groups are different, because the value of Levene's test is significant (p<0,05) (Levene, 1960).

In this case the means could be compared with Welch's d-test, which showed up a difference between the means (Welch, 1947), because the value of Welch's d-test is significant (p<0,05). It means the use of the game-based programming education had influence on the results of papers of the applied informatics engineering students.

4. Conclusion

I developed the game based programming education method until my 23 years long teaching period in Hungary at Obuda University and I tested successfully earlier at King Sigismund College (Kiss, 2013). My earlier experience showed some students have problems to follow the traditional way of teaching programing and they need an other point of view to find the key to pass this topic. I had the opportunity to test my developed method in a different country.

After the analysing process, we can say my starting hypothesis is correct; students get better paper results by using a game oriented programming education method and it is work independently from the country.

The applied informatics engineering students could take advantage of this learning method tool before the test. The students could get more than one mark better paper results when they took part in the game-based lessons.

The reason for this could be the fact that the game oriented lesson was preferred by the students who were motivated to write different type of dice or card game programs. Before the students need to know and use it well the basic program elements like selection, iteration and so on. The traditional way to teach programming for applied informatics engineering students was not so successful than the game-based examples. It means by same learning material and test exercises the student who learned programming in game-based method could have better paper results. The presented game-based teaching method is successful in Slovakia at J. Selye University too.

Acknowledgements

References

Anderson, L. W.; Krathwohl, D. R. and Bloom, B. S., (2001). A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives. New York ; London: Longman

Bloom, B.S.; Engelhart, M.D.; Furst, E.J.; Hill, W.H. and Krathwohl, D.R. (1956). Taxonomy of educational objectives - the classification of educational goals. Handbook 1: cognitive domain, B.S. Bloom, Ed. New York: Longmans, Green

Cohen, J. (1973) Eta-squared and partial eta-squared in fixed factor ANOVA designs. Educational and Psychological Measurement, 33, (pp: 107-112)

Ersoy, Y. (2005). Matematik eğitimini yenileme yönünde ileri hareketler-I: Teknoloji destekli matematik öğretimi. The Turkish Online Journal of Education Technology, Volume 4, Issue 2, 2005, ISSN: 2146 – 7242, (pp. 51-63)

Esper, S., Foster R. S., Griswold G. W. (2013), On the nature of fires and how to spark them when you're not there. SIGCSE '13 Proceeding of the 44th ACM technical symposium on Computer science education (pp. 305-310)

Gastelú, C. A. T. (2013). Digital Skills in Mexican Children, Advances in Intelligent Systems Research Journal, Volume 26, ISSN 1951-6851, pp: 1162-1165, published by Atlantis Press, doi:10.2991/icibet.2013.155

Hoffmann, M.H.W. (2011). When to Assess Knowledge, Skills and Competences. 22nd EAEEIE Annual Conference, Maribor, Slovenia, 2011, IEEE Catalog Number: CFP1196D-ART, ISBN: 978-961-248-281-7, (pp. 1-5)

Ismail, Mohd Nasir; Ngah, Mohd Nasir; Umar, Irfan Naufal (2010). Instructional strategy in the teaching of computer programming: A need assessment analyses. The Turkish Online Journal of Education Technology, Volume 9, Issue 2, 2012, ISSN: 2146 – 7242, (pp. 125-131)

Isman, A. and Celikli, G. E. (2009). How Does Student Ability and Self-Efficacy Affect The Usage of Computer Technology, The Turkish Online Journal of Educational Technology, ISSN: 1303-6521, volume 8, Issue 1, (pp: 33-38)

Kiss, Gabor (2010a). Using the Lego-Mindstorm kit in German Computer Science Education. 8th IEEE International Symposium on Applied Machine Intelligence and Informatics (pp 101-104). IEEE Xplore digital library Digital Object Identifier: 10.1109/SAMI.2010.5423759

Kiss, Gabor (2010b). A Comparison of Programming Skills by Genders of Hungarian Grammar School Students. Symposia and Workshops on Ubiquitous, Autonomic and Trusted Computing, ISBN: 978-0-7695-4272-0, (pp. 24-30), IEEE Catalog Number: CFP1075H-CDR, IEEE Xplore digital library Digital Object Identifier: 10.1109/UIC-ATC.2010.83

Kiss, Gabor. (2012). Measuring Hungarian and Slovakian Students' IT Skills and Programming Knowledge. Acta Polytechnica Hungarica, Volume 9, No. 6, 2012, ISSN: 1785-8860, (pp. 195-210)

Kiss, Gabor. (2013). Teaching Programming in the Higher Education not for Engineering Students, Procedia of Social and Behavioral Science Journal, Volume 103, ISSN: 1877-0428, (pp. 922-927)

Kurland, D. Midian; Pea, Roy D.; Clement Catherine; Mawby, Ronald. (1986). A Study of the Development of Programming Ability and Thinking Skills in High School Students. Journal of Educational Computing Research, Vol. 2, Issue 4, 1986, ISSN: 0735-6331, pp. 149-

Levene, H. (1960). Robust tests for equality of variances : In Ingram Olkin, Harold Hotelling, et alia. Stanford University Press, (pp: 278–292)

Welch, B. L. (1947). The generalization of "Student's" problem when several different population variances are involved. Biometrika 34 (1–2), pp: 28–35