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.

Keywords: game-based, oriented, programming, education, teaching

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