

# Methodology of Group Work Organisation for Student Learning Performance Improvement

**Natalya Prokofyeva, Marina Uhanova**

*Riga Technical University, Faculty of Computer Science and Information Technology.*

*Address: Seta str. 1, Riga, LV-1048, Latvia.*

**Abstract.** *The article describes the methodology of learning programming for students of various engineering disciplines. The course "Algorithmization and Programming of Solutions" is taught to all first-year students of the Faculty of Computer Science and Information Technology in Riga Technical University and provides the basic knowledge of the principles of computational process algorithmization and software creation technology using Java programming language. There are 8 laboratory assignments in the course, where students have to develop a software programme and 2 group work assignments, where the student has to develop some algorithms to solve a given problem, write a programme, evaluate the speed of developed algorithms and prepare a presentation on the results of their research. The article describes the main principles of efficient student group work organisation that lets to increase their interest and motivate them to participate in the course in a responsible way. This paper is focused on research on how group work influences student learning performance.*

**Keywords:** *Java, student group work, survey, results.*

## I. INTRODUCTION

Nowadays, due to the increasing pace of scientific and technical progress, requirements to knowledge and skills of modern specialists continuously grow. As the result, need for quality education of specialists increases as well. However, the discrepancy between the capabilities of traditional teaching methods and the amount of actual knowledge that modern society demands from graduates of educational institutions indicates a problem in the system of modern higher education. Therefore, the issues of computer training and knowledge control are of interest for many researchers, in the field of both education and information technologies [1][2][3].

Usually, the solution of routine tasks requiring large amount of computations is assigned to computers. Although there are many programmes for solving similar tasks, computer cannot compete with human in solving creative problems. Many international scientific conferences (such as IEEE ICALT, IADIS e-Learning, IASTED CATE, etc.) and e-journals [4][5][6] are dedicated to research in the field of computer-based learning and knowledge assessment. Therefore, the need to teach future engineers the basics of programming is undeniable.

As is known, every learning course comprises three components: the main (theoretical) part (ideas and knowledge); the laboratory-practical part (skills, experience); the evaluation part (evaluation of skills).

All these parts are obligatory and are traditionally taught by all teachers in higher education [7].

In the papers [8][9][10], results of influence of computer-based testing of knowledge on the learning performance of students were described.

Attendance of all the test lessons and use of computer systems for training and evaluation of knowledge (Learning Management system, ORTUS and IKAS systems) promoted more successful fulfilment of students' practical assignments and passing an examination.

The authors of this paper also assume that the addition of substantial new components to the course "Algorithmization and Programming of Solutions", specifically the addition of group projects, can significantly raise the motivation of students, their interest in improving their own professional competence and quality of their education.

## II. THE CONTENT OF "ALGORITHMIZATION AND PROGRAMMING OF SOLUTIONS" COURSE

The course "Algorithmization and Programming of Solutions" is taught to all first-year students of the Faculty of Computer Science and Information Technology in Riga Technical University and provides basic knowledge of the principles of computational process algorithmization and software creation technology using Java programming language [11]. In this course, several practical works

are envisaged. Organization of practical tasks takes place in the following way. Each student must develop an algorithm, write a program and submit it electronically to the study portal ORTUS. Once the program gets evaluated, the student must defend his or her work, i. e. write a report and answer teacher's questions about the program and the work in general. There are 8 laboratory assignments in the scope of the subject, where students have to develop a software programme. The first part of the course includes five (Branched programs; Development of a simple cyclical program; Processing one-dimensional arrays; Processing two-dimensional arrays; Ways of organization of nested loops), the second one – three laboratory works (Sorting arrays; Lines and text files; Creation of a file processing system).

Success of the mastering of bases of programming strongly depends on the acquisition of experience during independent solving of practical problems.

To increase interest and to motivate students to take responsibility for their own understanding of course materials, during the academic year students are offered to perform two group projects, one project in each semester. Each group project consists of developing a program to solve some problem and doing research on the efficiency of chosen algorithms for this solution.

### III. THE CAUSES AND IMPORTANCE OF THE WORK IN GROUPS

The general scheme of group work is presented in Figure 1.

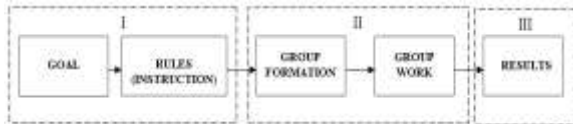


Fig. 1. Group work scheme

The basic purpose of the group project is to provide students an opportunity to exchange the acquired experience (Fig. 1, I stage). At this stage, the participants of groups receive assignments and deadlines, as well as criteria for evaluating the results of their planned research.

Students are allowed to freely divide into groups of 3 to 5 participants, and each one has to develop from one to two various algorithms for the solution of the offered task (Fig 1, II stage).

Once the program is developed, students must investigate empirically the speed of the chosen algorithms and the developed program and prepare a presentation on the obtained results. Students must also prepare a speech on the results of research at one of the practical lessons.

In the third stage (Fig. 1, III stage), all members of the group receive the same mark for the work they have accomplished. By student vote, the best presentation is selected. The authors of that presentation are awarded an additional point.

In the first semester, it is offered to students to solve the following problem as a group project. It is necessary to develop a program that would find a way through a maze. The description of the maze is stored in a two-dimensional array, whose elements are equal to zeros and ones (1 – a wall, 0 – pass). The entrance of the maze is the first element of the array (with indexes [0][0]), and the exit is the last element of the array (with indexes [R-1][K-1], where R is the number of rows and K – the number of columns in the array). For example, the description of the maze can look as showed in Figure 2.

|   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Fig. 2. Maze description

Thus, the program should do the following:

- input from keyboard the size of the two-dimensional array containing the description of the maze (numbers of rows and columns);
- input from keyboard the elements of the array;
- output the information about the developed algorithms and allow the user to choose one of them;
- if a path exists, then output it in the following format: (0,0) (1,0) (1,1) (1,2) (1,3) (2,3) ... (6,6).

Each member of the group is required to develop at least one algorithm for searching the path through the maze. Thus, the developed program must be able to find a path in the maze using at least three different algorithms.

Students must assess all of the developed algorithms in terms of passage speed and path length (if there are multiple paths in the maze), as well as determine, how the size of the maze influences the speed of the algorithms used in the programme. Students must present their solutions on one of practical lessons.

In the second semester, students are involved in developing and comparing various methods of array sorting. Students are offered to investigate at least six of the following methods: selection sort, insertion sort, bubble sort, odd-even transposition sort, merge sort, heap sort, counting sort, Shell sort, radix sort, tournament sort, quick sort. They are allowed to develop and investigate any other methods described in study literature or on the Internet.

In general, the second group project task is formulated as follows. 1) develop a program that inputs count of elements and array from keyboard, sorts the array in ascending order and outputs the array to the screen. Implement at least six different

sorting methods in the program. 2) Using the developed program, empirically determine the relation between program execution time and the amount of elements in the array (to construct the relevant tables and diagrams). 3) Make recommendations, which method is best used, if:

- the array contains a small amount of items;
- the array contains a big amount of items;
- the items of the array are arranged in reverse order;
- most items of the array are arranged.

To evaluate the effect of group projects on student learning performance and to improve the organisation student group work, a student survey was conducted. The results of this survey are presented in the following chapter.

#### IV. SURVEY CONTENTS AND RESULTS

The aim of the survey was to identify the principal reasons for participation or non-participation in group projects, because participation in group projects was not obligatory for all students. It should be noted that during the past two years 187 or 23% of the total number of students participated in group projects.

To increase the number of students participating in group projects, students were asked to select the main reasons preventing them from participating in a group project or successfully completing it. As the main reasons the students selected such factors as lack of time (74%), excessive complexity of the offered tasks (22%), difficulty in organising group (28%), unwillingness to work in a group (24%), unwillingness to prepare presentations and speeches (24%). The results of the survey are presented in Figure 3.

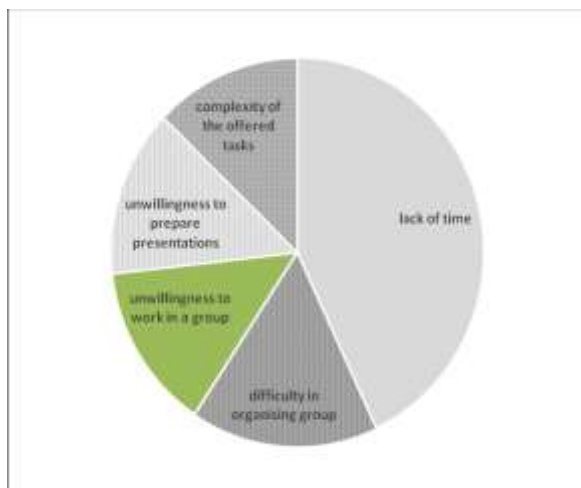


Fig. 3. Main reasons preventing students from participation in group projects

During group projects implementation, the students have more often faced such problems as lack of time (13%), refusal of some members of the group to do their part of the task (3%), difficulties in communicating with other members of the group

(4%), difficulties in finding time and place for teamwork (19%), difficulties in preparation and presentation of the obtained results (6%) (Fig. 4)

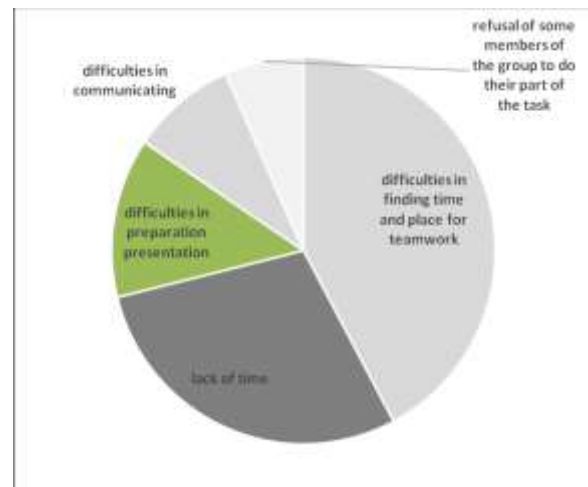


Fig. 4. Main problems preventing students from successfully completing their group projects

The survey showed that 77% of students with no prior group project experience are ready to participate in the new group project if there will be such opportunity. Among those who have participated in at least one group project, 99% are ready to do it once again.

As the main reasons of participation in a group project, students have indicated the wish to learn something new (13%), interesting tasks to solve (19%), as well as the wish to get a higher mark (25%) (Fig. 5).

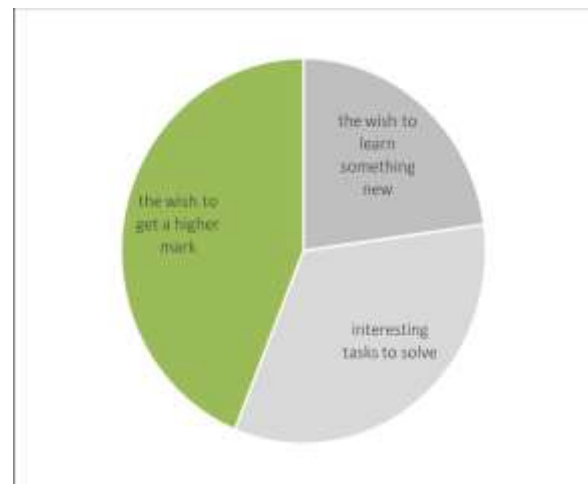


Fig. 5. Main reasons of student participation in group projects

During the survey, the students also noted that participation in a group project has allowed them to improve their programming skills (25% of students stated so), prepare presentations and speeches (11% of students), improve communication skills and learn to work in a group (15%), as well as find new friends (4%). At the same time, 6% of students have

indicated that the only benefit of their participation in the group project was a higher mark (Fig. 6).

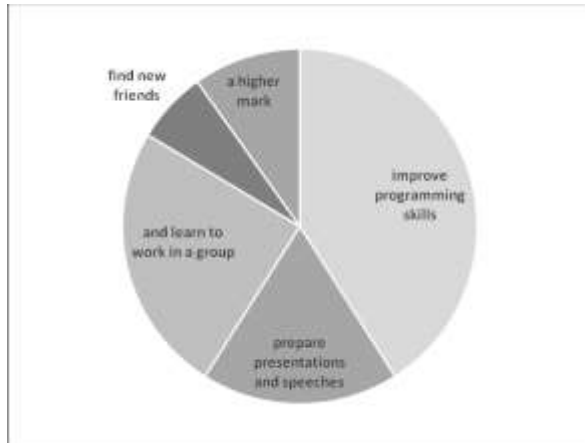


Fig. 6. Results of participation in group projects

### V. CONCLUSION

Since the methodology of the group work is used only during last two years, there is not enough statistical data to precisely describe the relationship of work in groups and student learning performance. But the results of the survey have showed that 24% of students who have participated in group projects, have received very good (8) or higher at the examination, while only 12% of the students who did not participate in any group project have received the same high marks at the examination.

In the future work, it is planned to collect statistical data rich enough to mathematically determine the influence of group projects on the student's learning performance, as well as to develop more engaging and effective group tasks.

### REFERENCES

- [1] Howard L, "An exploration of autonetnography as an eResearch methodology to examine learning and teaching scholarship in Networked Learning", *Electronic Journal of e-Learning*, Vol. 14, Issue 5, 2016, pp. 332 – 335.
- [2] M. Kryshchuk, J. Lavendels, V. Sitikovs. Models of data and their processing for introductory courses of computer science. No: Proceedings of the 10th International Scientific and Practical Conference "Environment. Technology. Resources", Latvia, Rezekne, 18.-20. June, 2015, 134.-137.lpp. Available from: DOI: <http://dx.doi.org/10.17770/etr2015vol3.178>
- [3] Lau, P. N. K., Lau, S. H., Hong, K. S., & Usop, H., "Guessing, Partial Knowledge, and Misconceptions in Multiple-Choice Tests", *Journal of Educational Technology & Society*, 14 (4), pp. 99–110.
- [4] The Electronic Journal of e-Learning (EJEL) [Online]. Available: <http://www.ejel.org/main.html> [Accessed: March, 2017].
- [5] *Journal of Educational Technology & Society* [Online]. Available: English version: <http://www.ifets.info/>, Russian version: <http://ifets.ieee.org/russian/periodical/journal.html> [Accessed: March, 2017].
- [6] International Academy, Research, and Industry Association (IARIA) [Online]. Available: <http://www.iariajournals.org/> [Accessed: March, 2017].
- [7] Prokofjeva, N., Uhanova, M., Zavjalova, O., Katalņikova, S. Structuration of Courses at Studying Disciplines of Programming. No: Proceedings of the 10th International Scientific and Practical Conference "Environment. Technology. Resources", Latvia, Rezekne, 18.-20. June, 2015, 159.-163.lpp. ISBN 978-9984-44-173-3. ISSN 1691-5402. Available from: doi:10.17770/etr2015vol3.179
- [8] N. Prokofjeva, D. Zagulova, S. Katalņikova, "Possibilities of Use of Computer Systems in Educational Process by Preparation of IT of Experts," *Educational Technology & Society*, 2013, Vol.16, No.4, pp.511-520. ISSN 1436-4522. [Electronic resource] / [http://ifets.ieee.org/russian/depositary/v16\\_i4/pdf/18.pdf](http://ifets.ieee.org/russian/depositary/v16_i4/pdf/18.pdf) [Accessed: March, 2017].
- [9] Zaiceva, L., Prokofjeva, N. Knowledge Control during the Preparation of IT Specialists in Riga Technical University. In: Proceedings of International Congress on Information Technologies (ICIT-2012): International Congress on Information Technologies (ICIT-2012), Russia, Saratov, 6-9 June, 2012. Saratov: Saratov State Technical University, 2012, pp.1-10.
- [10] N. Prokofjeva, A. Anohina-Naumeca, O. Lebedeva, Administration of Knowledge Assessment at Riga Technical University. Proceedings of the 8th International Multi-Conference on Computing in the Global Information Technology (ICCGI 2013), July 21-26, 2013, Nice, France, IARIA, pp.34-39, ISBN: 978-1-61208-283-7.
- [11] Prokofjeva, N., Uhanova, M., Katalņikova, S., Synytsya, K., Jurenoks, A. Introductory Programming Training of First Year Students. *Procedia Computer Science*, 2017, Vol.104, pp.286-293. ISSN 1877-0509. Available from: doi:10.1016/j.procs.2017.01.137