Diversifying Activities to Improve Student Performance in Programming Courses

Scheila Wesley Martins
Pontifícia Universidade Católica de Minas Gerais

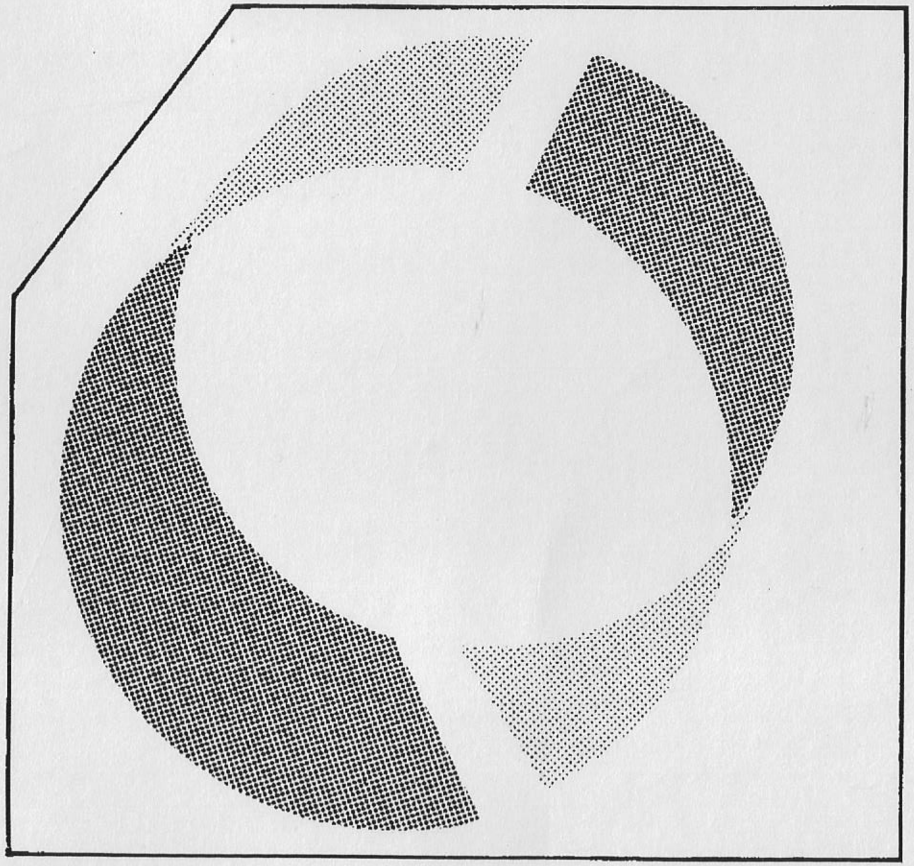
Antonio Jose Mendes
University of Coimbra

Contents

Communication & Cognition

Vol 46 --- Number 1-4 --- 2013

Volume 46, Number 1-4, 2013



COMMUNICATION & COGNITION

Communication & Cognition
Vol. 46, No. 1 - 4 (2013), pp. 39 - 58

DIVERSIFYING ACTIVITIES TO IMPROVE STUDENT PERFORMANCE IN PROGRAMMING COURSES

Scheila Wesley Martins, Antonio Jose Mendes, Antonio Dias Figueiredo

Abstract

This paper presents a pedagogical strategy that intends to help students maximize learning and minimize drop-out rates in programming courses. The goal is to motivate students to develop a better programming study behaviour through the utilization of appropriate learning activities and the conscious assessment of their self-efficacy level. The paper also includes some preliminary results of the strategy application with students of Design and Multimedia.

Keywords

Programming learning, Research Communities, Self-efficacy.

1. Introduction

There is an intense effort from researchers and teachers worldwide to understand the reasons that make programming learning so difficult for many students [Lahtinen, Ala-Mutka, and Järvinen, 2005]. In fact, it is common to find students that experience many difficulties to develop problem solving competences, and to use those competences to create programs that solve basic problems. It is important to make students realise that programming is, above all, a conscious exercise of mental abilities that can be developed through adequate activities and specially through effort.

One pedagogical strategy directed to programming learning should make students aware that solving programming problems is an activity that they are fully capable of accomplishing. It is important to value contexts and establish class dynamics that may motivate students to teamwork, giving evidence and convince them that individual difficulties can be solved if they get ready hard work and to "learn to think". This should lead to a higher student commitment to their learning, including behavioural changes that may improve their performance throughout the course.

In the next sections we propose a strategy designed with the above objectives. We also describe some of the theoretical inspirations, tools used, and the results obtained in a Programming course at the University of Coimbra.

2. High Education Reform

Access to education is an evident concern of today's society and the global economy, with a strong commitment from governments of rich and developing countries to initiatives to improve and offer better conditions for learning at the different stages of the educational system. Nevertheless, those reforms have not yet produced a system that truly develops the "teaching to think" concept.

Nowadays many students, from elementary to high school, do not develop the different skills and competences necessary sufficiently throughout their university years, as the results of International Programs of Teaching Evaluation show [Dohn, 2007]. The growing number of university students has burdened the traditional academic model. In a short timeframe there has been a growing demand that made the academia change their teaching models, and without any prior preparation.

Faced with the urgency to deal with this situation, academia often choose processes that privilege administrative issues (class format, resource allocation), which is often not the best didactic solution. Although there is a need for a renovation, sometimes administrative reforms do not focus enough on didactic matters, and that tends to end up hindering the evolution of didactic processes [Martins, Mendes and Figueiredo, 2010].

3. Motivation and Learning

Motivation has a great impact on the individual's cognitive development and is a determinant factor for success in a learning process. Maslow's Motivation Hierarchy [Maslow, 1954] has originated many research works. Understanding the motivation to learn requires a profound analysis of the socio-cognitive components of personality (identification with the: institution, degree, career, accessibility to teaching and the analysis of academic success) and the quality of relationships in coexisting environments (social identification and learning approaches) [Abreu, 2002].

Throughout the formal learning process, from primary school, the student faces various tasks, contexts and learning methodologies. From those experiences the students develop their behavior, discovering a learning strategy and methodology that best fits their personality, beliefs and values. It is impossible to understate the impact of the learning methodologies experienced, as they greatly influence all aspects of the competences that a generation of students acquires.

The quality of those skills and competences often has a close relationship with how stimulating and motivating the role learning process is. Thus, the motivation aspects in the classroom are something that could be as important as the content, which could turn the knowledge acquisition into an unstoppable and enjoyable journey, or not.

Today we often see students in classes simply quitting from trying to solve some problem, simply because they don't see a solution immediately or their first attempt does not work as expected. We do believe that students often choose to drop out when they lose trust on their own abilities. So, as important as the content, some motivation strategies must be developed within the course's syllabus and used if we want to build a stimulating classroom which will give us better ways to help students start improving the academic skills they need to overcome their natural difficulties [Roberts, 2000].

Specially important in preventing dropping out behaviour, is that teachers know about the motivations issues, and are able to discover when and how to perform intervention actions in classroom dynamics.

4. Teach to think as an inspiration

Matthew Lipman is an educational theorist who called the attention of the North-American Academic Community in the seventies with his proposal of teaching philosophy to children, through the "Pedagogy of Judgment" (Lipman, 1991). It suggests "teaching to think" using philosophical speeches and proposes that teachers readopt Socratic teaching as a didactic approach.

His goal is to start a long-term process of development of critical and creative thinking in infancy, combining literacy and language acquisition, lasting throughout all the child's formal educational process. While developing literacy skills and their ability to think begins a lifelong development.

To Lipman, making judgments is the basic cognitive unit for the development of such thinking, and it is influenced both by criteria (its rational element) and by individual values (the emotional element). His proposal defines practical actions, which converge with the thinking of John Dewey, Lev Vygotsky and Jerome Bruner, aiming at changing the erroneous concept that childhood innocence impedes the child from learning to use reasoning as a learning tool. His work shows that the lack of motivation to learn must be understood more as a result of the evolution of non-reflective practices of the traditional educational model, rather than an "innate" lack of curiosity from the student.

Although Lipman's approach is originally used by English teachers in K-12 classroom, his approach could be understood as an abstraction about a way to give teachers tools to stimulate students to learn "how to think" as a study methodology. Not just that, but also how to do research and why they must value the process to reach knowledge, after all this is a kind of thing they will and must do for the rest of their lives.

Lipman's proposal is for teachers to focus their practices on guiding students in a search for knowledge, motivating them to learn how to consciously identify what they already know and what they need to know. It is important not only to solve a problem, but also to be able to think about and assess the quality of a solution.

Learning programming has nuances similar to the context of language acquisition by children [Wells, 1999], and resembles the relationship between language and the programming paradigm. That is why linking Lipman's approach to learning programming is rather useful and relevant.

5. Teaching To Think In Programming

Our goal is to identify the characteristics of contexts that may make programming learning more stimulating, minimize drop-out intentions and make students learn more and better. The pedagogical strategy we propose includes a set of guidelines regarding contexts and didactic activities, computational tools and motivational measures that may assist teachers in the definition of specific learning contexts for programming courses, as presented by Fig.01.

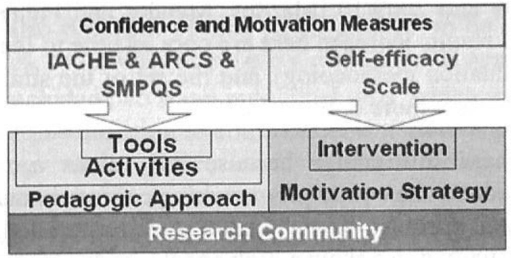


Fig. 1. Pedagogical Strategy

The proposal was developed under the perspective of learning communities, inspired by a metaphor of Mathew Lipman's communities of inquiry [Lipman, 1991], considered to be a relevant abstraction for proposals involving the development of critical thinking and literary skills, and also as a strategy to improve the capacity to solve programming problems among university students [Martins, 2005].

The course context should include didactic activities planned to strengthen the student's involvement with the process of knowledge acquisition and development of competences to solve problems, through teamwork and the motivation to practice their literary skills in several ways, such as: collaborative knowledge production through small projects and research activities, peer tutoring and continuous assessment.

The context can include computer tools that might help learning, such as algorithm simulation or software to support competitions and testing of programs. To stimulate extra-curricular activities and to facilitate monitoring and continuous assessment tasks, it is important to use a Learning Management System. Motivational measures should be checked regularly to support student guidance and to adequately direct the teacher's efforts in student motivation and the prevention of behaviours that may lead to students dropping out.

6. The Experiments description

Our proposal was experimented with students enrolled in a Programming course, part of the Masters Degree in Design and Multimedia (MDM) at the University of Coimbra, in the academic years 2008/09 and 2009/10 [Martins, Mendes and Figueiredo, 2010]. However, the results included here are only relative to the second year, since the evaluation methodology and the set of the strategy activities have evolved into a more definite format.

We chose this course because it involves a much smaller number of students than other programming courses in our department, which may easily present more than 300 students enrolled, and because its students usually don't show a high appetence for programming. The small number of students permitted a close student monitoring. This allowed the teacher to know well the students, and to adapt the class dynamics to a research approach during group based problem solving.

Our strategy doesn't make a clear distinction between theoretical, practical or lab classes, something that occurs in many programming courses in our department. All classes are spaces for knowledge construction and practical experimentation, making up a total of 6 weekly hours of work. Bearing in mind the artistic background of the involved students, we chose to create a context based on visual hands-on projects of growing complexity, as it would facilitate the students' involvement and interest in the activities.

We used the programming language Processing [Greenberg, 2007] as it facilitates the development of artistic works, keeping the power of Java language. We also used Moodle platform as a basis for some activities. The course was mostly based on practical learning. The visual issues' around the exercises and projects proposed involved a need for research, especially the review of algebra and trigonometry knowledge. We used several types of activities during the semester with specific objectives:

- Individual seminars on artistic projects developed in Processing were used to raise students' interest and motivation about programming;
- In specific moments we used individual challenges, inspired in JiTT challenges [Bailey and Forbes, 2005], as a way to stimulate individual work, especially outside the classroom. These challenges included a self-evaluation component, making the student used to critical assessment;
- We also proposed several small projects to be developed in groups, followed by discursive evaluation of peers' work;
- We included two small individual tests, which were preceded by a test simulation to allow students to have a more concrete feeling about their level, without being under real assessment;
- Finally students were asked to create a portfolio including their own programming projects and other related materials;
- All these activities were evaluated by the students in biweekly reflections about their satisfaction with their own performance, tasks, materials and class's rhythm.

This experiment took place between September 2009 and February 2010. The course had 18 registered students, although only 15 really got involved in it. Most students were recent graduates in the areas of arts and design, but two were Polish Erasmus students from Physics Department and weren't considered in the statistics analysis.

7. The evaluative tools

The pedagogic strategy in progress had different ways to be assessed and nowadays includes some psychological formal instruments used to evaluate two different things: 1) to assess the experiments results and 2) to assess the learning process.

For the first kind of assessment we used the Inventory of Attitudes and Study Behaviours (IABS/IACHE) [Monteiro, Vasconcelos and Almeida, 2005] to get information about study behaviour, the Course Interest Survey (CIS) [Keller, 2009] to measure students' motivation according to ARCS model. For the second kind we used the Student Motivation Problem Solving Questionnaire (SMPSQ) [Margolis, 2009] to assess the level of satisfaction with the different learning activities and a self-efficacy test [Ramalingam and Wiedenbeck, 1998] to keep students alert regarding the quality of their learning.

The IACHE is a survey developed by Portuguese researchers, and it is an independent, generic behavioural test that might evaluate important aspects of the cognitive and motivational measures about the students' college behaviour. The IACHE encompasses cognitive, motivational and behavioral aspects, distributed in five sub-scales:

- Comprehensive focus, using reflection and deep content analysis, which implies an higher effort and time in learning;
- Reproductive focus, the tendency to spend only a minimum effort on a superficial learning, based on memorization and content reproduction;
- Competence personal perception, a measure on how students see their own competence in the course;
- Involvement, or motivation, related essentially with intrinsic motivation; and
- Organization, analyzes the indications of some level of organization on study activities.

With the advance of research in Distance Education (DE), some motivation evaluation tools focus on more specific features, such as the case of the Course Interest Survey (CIS), inspired by John Keller's Motivation Model ARCS (Attention, Relevance, Confidence and Satisfaction). It may show the levels of attention, relevance, confidence and satisfaction among students regarding a given course or e-learning environment, assess aspects such as pedagogical approach, class rhythm, teaching practice and proposed activities.

Identify the level of resistance to certain tasks may help evaluate motivational factors and make ease the task of understanding what lies behind many learning difficulties. The Student Motivation Problem Solving Questionnaire (SMPSQ) was developed as a strategy so that teachers and parents have a way of making a student express their expectations concerning success and failure, and the time and energy they are willing to invest in concluding a school task.

Besides, we also want establish a measure of motivation more closely associated with the aptitude of students to learn to program, which would also allow us to be measured totally independently from the motivational models observed. That measure is self-efficacy [Bandura, 1977; Margolis and McCabe, 2006] evaluated from scales directly associated with a self-analysis of individual ability or inability to perform a specific task. A self-efficacy scale for programming is a formal tool that may be independently and regularly used, and which provides a self-assessment of learning, as opposed to traditional grading systems. The scale used for Processing was translated and adapted from a scale for Java [Askar and DaVenport] since Processing is an integrated development environment (IDE) to for Java language.

8. Tests Results and analysis

In each survey the questions were answered by students according to the intensity of their level of accordance: from 1 (means no, totally false, or totally unconfident) until 5, 6 or 7 depending on each survey (means yes, totally true or totally confident), the answer 0 (means don't know) is possible only in SMPSQ survey.

The score for each aspect assessed is given by the sum of the answers to the corresponding question. The surveys' structure and their reference values (minimum, maximum and average point) are summarized in table 1. There was another analysis carried out, called intensity levels, also show in table 1. It was proposed to identify change on students' answers patterns, by assembling the groups of answers in three levels: low, medium and high.

	Tests Composed		Statistic Measures			Intensity Levels of Answers		
	Questions	Answers	Min	Max	X_m	Low	Medium	High
IACHE	44	1-6	10 [a] 8 [b]	60 [a] 48 [b]	35 [a] 28 [b]	1-2 [c]	3-4	5- 6 [c]
CIS	34	1-5	9 [d] 8 [d]	45 [d] 40 [d]	27 [d] 24 [d]	1-	3	4-5
SESP	32	1-7	32	224	128	1-2	3, 4, 5	6-7
SMPQS	15 5	0 [h], 1-5 0 [h], 1-5	15 [f] 5 [g]	75 [f] 25 [g]	45 [f] 15 [g]	1-2	3-4	5-6

(a) For comprehensive focus and organization dimensions; **(b)** For reproduce focus, involvement and competence personal perception dimensions; **(c)** For competence personal perception dimension, the low level is 5 and 6, and the high level is 1 and 2; **(d)** For relevance and satisfaction aspects; **(e)** For attention and confidence aspects; **(f)** For part I - motivation about academic tasks; **(g)** For part II - general motivation; **(h)** When it appears means that more research has to be done about the question (Don't Know level);

Table 1. Test Data Specifications Summary

Given the qualitative nature of data, we used the Wilcoxon Signed-Rank test, a non-parametric statistic test, to analyse it. We also carried out a statistical comparison between the average results, and the analysis of the percentage of replies arranged in scales of intensity.

Table 2 presents the results in each dimension of IACHE, in columns I to V, and in the self-efficacy test, in column VI. In both cases, pre-tests were made in the beginning of the course and post-tests in the final part. The table includes average answers and the percent of students who gave answers in each of the three intensity levels.

	PRE						POST					
	I	II	III	IV	V	VI	I	II	III	IV	V	VI
Average	42.6	29.9	21.5	36.6	31.4	114.0	39.9	28.1	28.1	33.1	31.4	127.6
Low(%)	4	20	9	3	23	24	5	23	10	4	31	11
Mean(%)	52	47	34	42	59	62	69	52	61	64	55	75
High(%)	44	33	57	55	18	14	26	25	29	32	14	14

Comprehensive Focus-I, Reproduce Focus-II, Personal Perception-III, Involvement-IV, Organization-V, and Processing's Self-Efficacy-VI

Table 2. IACHE and Self-Efficacy Means and Intensity levels Summary

When comparing IACHE averages in pre and post-test, we can see they have decreased, except for Personal Perception (III) that increased, and Organization (V), that had no change. However, the averages of the comprehensive focus in both tests were higher than the average of the reproductive focus. This is a good sign that may have resulted from the stability verified in the organizational dimension. We believe that the inclusion of the challenges in the pedagogical strategy demanded an organizational effort from the students, especially outside classroom, so that they could meet the different deadlines.

These results can be considered positive, even though there was a small decrease in the involvement and comprehensive focus. The same can be said about the analysis by level of intensity, as we could see some migration of answers from low level to medium level, although some also migrated from high level to medium level.

The results of the Wilcoxon Signed-Rank test for IACHE test are displayed in table 3. They indicate that the decreases in comprehensive and reproductive focus averages cannot be considered statistically significant. Even the result of the involvement focus ($\rho = 0.061$) is very close to the statistical limit defined for this type of test ($\rho = 0.05$), which doesn't give a strong support to prove a statistic relevant difference. On the contrary, the variations on personal perception and organization are statistically relevant ($\rho = 0.021$), which suggests some modification of the students' behaviour during the course.

	I	II	III	IV	V	VI
Negative Ranks (Pos < Pre)	7	10	2	9	8	3
Positive Ranks (Pos > Pre)	2	2	8	2	2	9
Ties (Pos = Pre)	3	0	2	1	2	0
ρ-Value	.085	.134	**.021**	**.061**	**.021**	**.021**

Comprehensive Focus-I, Reproduce Focus II, Personal Perception-III, Involvement-IV, Organization-V and Processing's Self Effcacy-VI

Table 3. Wilcoxon range test summary

We had expected to find an increase in the averages of involvement and comprehensive focus. However, we found a slight decrease, which means that students didn't develop as much as we expected in those dimensions. As positive points we noticed the stability in organization and the decrease in the reproductive dimension, which means that students understood that this dimensions isn't particularly relevant for programming learning. The most negative aspect was the increase in personal perception, since it means that the students' level of trust in their own skills to be successful in programming learning decreased. This is quite worrying as students

with low expectations tend to invest less effort in study activities and drop out more easily.

The results obtained with the self-efficacy scale, column VI on table II, were more positive, as they revealed a positive evolution in students' confidence on Processing as showing by Fig. 2.

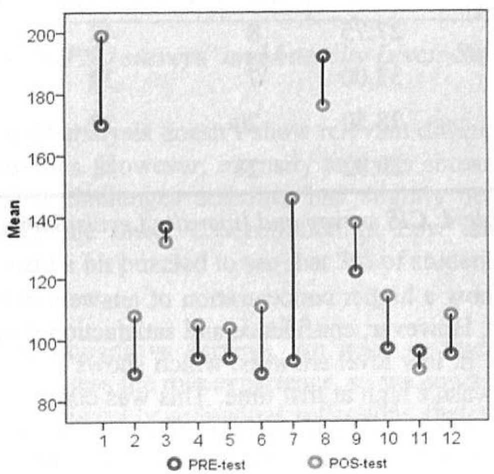


Fig. 2. Results for Processing Self-Efficacy scale 10

The Wilcoxon test result, column IV in table 3, ($\rho = 0.021$) also proved that there was a relevant difference between pre-test and post-test results. The analysis of intensity levels revealed that the number of low level answers has decreased, migrating to the medium level, while the concentration of high level answers remained unchanged. We also see that in the pre-test a little over 25% of the sample already presented individual scores higher than the medium point reference value. In the post-test only 25% of the sample decreased the value of its score, which means that 75% of students kept or increased their self-efficacy level for programming using Processing.

We used the CIS survey to measure the motivation levels, according to the ARCS model, aiming to discover how much relevance, motivation, confidence and satisfaction the students presented in the middle of the course. This could give important information to the teacher, detecting situations that might require his intervention, either with a particular student or with the whole course. The results can be seen in table 4.

CIS Dimension	Mean	Low (%)	Mean (%)	High (%)
Attention	27.75	8	43	49
Relevance	33.00	7	32	61
Confidence	28.50	20	28	52
Satisfaction	29.92	21	31	48

Table 4. CIS survey and Intensity Levels Summary

They show a higher concentration of answers in the high level, which is good. However, confidence and satisfaction dimensions show a higher value in low level answers, which shows that some students' level of trust wasn't high at that time. This was confirmed later by the rise in personal perception in IACHE. The results for attention and relevance dimensions showed that students were fully aware of the importance of the course and were consciously committed to work towards learning the necessary programming skills.

The SMPSQ test was used to identify the level of satisfaction and resistance felt by the students, specifically concerning the different activities proposed. The summary of SMPSQ's results is also presented in table 5. The test is divided in two parts, the first assess the motivation to perform a specific activity or task, and the second evaluates the reward expectations and the success to achieve the student's goals. The higher values obtained in the first part of the test, the less resistance or more motivated the student is. The same happens in part two, as higher values mean that the students have better personal perception levels for success. The statements answered with a 0 (Don't Know level) should be observed, as they may reveal causes for the students' resistance regarding a given activity and also possibly show their insecurity about their goals and success possibilities…

Activities assess by SMPSQ	Mean		DK (%)	Low (%)	Mean (%)	High (%)
	Part 1	Part 2				
Seminar	50.08	15.33	1	5	32	62
Code Analysis	49.80	17.50	3	10	41	46
Mini-test Simulation	51.07	16.69	2	8	39	51
Programming Challenges	40.30	13.46	1	6	43	50

Table 5. SMPSQ surveys' and Intensity Levels Summary

A statistical analysis doesn't show relevant differences between the various activities. However, intensity analysis shows that seminar and programming challenges activities had slightly better results. It was good to see the lower concentration in Low and DK levels. However, we were a bit puzzled to see that 3% of students' chose 0 for code analysis activities.

Besides the cognitive surveys, we want to find out how the students going to assess the role experience, so we conducted a content analysis over the biweekly individual reflections that students wrote during the course. The main goal was give them a safety communication channel with the teacher and a tool to express their evaluation, suggestions and concerns around the activities, the classroom rhythm and the strategy.

The results of student's reflections allow us to conclude that most negative aspects were related with students' past experiences and their fears of underperforming in the course, and not with the course itself. The positive and negative aspects mentioned by more students were identified. On the positive side the most representative statistical percents was:

- The motivational impact of class dynamics (93.33%);
- The high level of collaboration between students, the possibility to use their creativity in programming assignments, and teacher availability (86.67%);
- The good class rhythm, the individual support provided by the teacher, the learning activities, their performance in

challenges, and the improvement of their own study behaviour (80%);

- The sensation that their study effort was rewarded (66.67%);
- On the other hand, on negative side, we got:
- They recognized their lack of mathematical knowledge as negative factor which complicated learning (66.67%);
- They previous bad experiences in programming courses, they negative expectations about their own performance, the frustration about not being able to solve some problems, and the insecurity about their grades (60%);
- They complained about the amount of work and the course level of demand (40%).

The approval rates in a course are usually a good measure to assess the results of a pedagogical strategy, though this measure is sometimes overrated. When we think about motivation, the drop-out levels may be more important than final grades. Anyway, the evaluation of a particular strategy should include the students' results. In our case 80% of the students managed to pass the course, although most of them with average grades. Considering students' backgrounds and the difficulty associated with programming courses, we think the results obtained were good.

The teacher makes a positive evaluation of the strategy, not only due to the results obtained, but also considering the class dynamics. However, he also acknowledges a significant increase in his work, when comparing with more conventional approaches.

9. Conclusions and Future Work

The general evaluation of our experiment was considered positive by students and the course teacher. It may be considered an improvement over traditional approaches for the same course context. Both the course and the teacher's practice have received the highest scores in the final assessment of the course conducted by the Academic Services of the University of Coimbra, However, we recognize that some aspects weren't as positive as we expected. This means further reflection is necessary and some improvements will be made to the next edition of the course in the academic year 2010/11

The most relevant contribution to the present experiment was to suggest which activities and what kind of changes may be used to increase the motivation and contribute to a closer involvement of MDM students with learning programming. The statistics obtained do not allow us to see if the positive results will exist in students with other profiles. However, the enthusiasm and the good assessment of students and teacher makes us believe that the strategy used was determining for the results.

We also believe that experiments conducted and the results obtained contribute to reinforce the existing debate around the need to implement a reform in the context of the current programming classes model and other courses in the department, such as: high failure rates with up to 50% in some courses, classrooms overcrowding, which brings huge class management and resources issues.

We are aware that the changes suggested by the strategy may bring several implications: teachers' resistance in changing the daily didactic practice, students' resistance due to a more demanding class model, and the need for a reasonable financial investment to increase the number of teacher. This latest factor could overload the department's and University's budget. However, if we bear in mind that the current model also determines expenses that have not brought significant winnings in what concerns the quality and effectiveness of learning, the proposed investments may be justified.

We should also consider that the investment will not be eternal, if that it would imply a qualitative progress in programming learning, decreasing the drop-out rates. Bearing this goal in mind, some elements related with the strategy might be adapted to make it more easily implemented. This proposal was presented to the Administration and the Pedagogical Council of the University of Coimbra, that are now assessing the possibility of starting the process of changing the current model of programming learning, embodying these elements adapted from the strategy.

References

Abreu, M. V. (2002). *Cinco Ensaios sobre a Motivação.* 2nd ed. Livraria Almedina, pp. 57-96.

Askar, P. and DaVenport, D. (2009). "An Investigation of Factors Related to Self-Efficacy for JAVA Programming Among Engineering Students". *The Turkish Online Journal of Educational Technology (TOJET),* Vol. 8 (1).

Bailey, T. and Forbes, J. (2005). "Just-in-Time Teaching for CS0". *SIGCSE Bull.,* Vol. 37 (1), pp. 366-370. DOI= http://doi.acm.org/10.1145/1047124.1047469

Bandura, A. (1977). "Self-efficacy: Toward a Unifying Theory of Behavioral Change". *Psychological Review,* Vol. 84 (2), pp. 191-215.

Dohn, N. B. (2007). "Knowledge and Skills for Pisa-Assessing the Assessment". *Journal of Philosophy of Education,* Vol. 41, pp. 1-16.

Greenberg, I. (2007). *Processing: Creative Coding and Computational Art.* Friends of Ed Press.

Keller, J. M. (2009). *Motivational Design for Learning and Performance: The ARCS Model Approach.* (handbook), NY, Springer.

Lahtinen, E., Ala-Mutka, K. and Järvinen, H. (2005). "A Study of the Difficulties of Novice Programmers". *SIGCSE Bull.,* Vol. 37 (3), pp. 14-18.

Lipman, M. (1991). *Thinking in Education.* Cambridge University Press.

Margolis, H. (2009). *Student Motivation: A Problem Solving Focus* http://www.reading2008.com/Motivation Problem_Solving_Questionnaire-HowardMargolis-2009Jan1c.pdf

Margolis, H. and McCabe, P. (2006). "Improving Self-Efficacy and Motivation: What to Do, What to Say". *Intervention in School and Clinic,* Vol. 41 (4), pp. 218-227.

Martins, S. W. (2005). *Um modelo computacional de apoio ao desenvolvimento do pensamento crítico.* Master thesis. Brazil, Universidade Federal de Santa Catarina (UFSC).

Martins, S. W., Mendes, A. J. and Figueiredo, A. D. (2010). "A Strategy to Improve Student's Motivation Levels in Programming Courses". 40th Annual Frontiers in Education Conference, (Washington, DC, 27–30 Oct. 2010). FIE 2010, pp. F4F-1 - F4F-7. DOI: 10.1109/FIE.2010.5673366.

Maslow, A. H. (1954). *Motivation and Personality.* NY, Harper, 1st ed.

Ramalingam, V. and Wiedenbeck, S., (1998), *Development and Validation of Scores on a Computer Programming Self-Efficacy Scale and Group Analyses of Novice Programmer Self-Efficacy,* Journal of Educational Computing Research, Vol. 19 (4), pp. 367-81.

Roberts, E., (2000,). "Strategies for Encouraging Individual Achievement in Introductory Computer Science Courses". *SIGCSE Bull.,* Vol. 32 (1), pp. 295-299. DOI = http://doi.acm.org/10.1145 / 331795.331873.

Wells, G. (1999). "Language and Education: Reconceptualizing Education as Dialogue". *Annual Review of Applied Linguistics of Oxford,* UK, Vol. 19, pp. 135-155.

About the authors

PhD Student, **Scheila Wesley Martins**, Centre for Informatics and Systems (CISUC), University of Coimbra, Pólo II - Pinhal de Marrocos, 3030–290, Coimbra, Portugal, Phone: +351 239790078. She is a BSc in Computer Science by Universidade Federal de Mato Grosso (Brazil) since 1998, specialist in ICT area by Federal University of Lavras (Brazil) since 2001 and obtained her Master degree in Computer Science by Universidade Federal de Santa Catarina (Brazil) in 2005.

Her teaching experience in Computer Science has an emphasis on learning programming and algorithms and engineering education. As a researcher, she works on the following topics: learning programming, learning environments, critical thinking and midia competences. E-mail: scheila@dei.uc.pt.

Professor, **Antonio José Mendes**, PhD, Centre for Informatics and Systems (CISUC), University of Coimbra, Pólo II - Pinhal de

Marrocos, 3030–290, Coimbra, Portugal, Phone: +351 239790036, E-mail: toze@dei.uc.pt. He holds a PhD in Electrical Engineering specialization in Computer Science, by the University of Coimbra, since 1996.

He is an Assistant Professor at the Department of Informatics Engineering (DEI in Portuguese) of the Faculty of Science and Technology of the University of Coimbra, where he teaches courses in the area of Programming.

He is a member of CISUC, coordinating the activities of the Educational Technology Lab, part of the Cognitive and Media Systems Group. In this context he develops research in ICT in Higher Education, with particular emphasis on the use of technology to support learning programming.

Professor, **António Dias Figueiredo**, PhD, Centre for Informatics and Systems (CISUC), University of Coimbra, Pólo II - Pinhal de Marrocos, 3030–290, Coimbra, Portugal, Phone: +351 239701266. He obtained his Ph.D. in Computer Science from the University of Manchester, U.K., in 1976.

Between 1984 and 2007 he was professor of Informatics and Information Systems at the Faculty of Science and Technology of the University of Coimbra, where he created, in 1994, the Department of Informatics Engineering.

He is emeritus professor of Information Systems of the Faculty of Science and Technology of the University of Coimbra, Portugal, where he develops his research interests in Business Information Systems and ICT in Learning and Education at CISUC. He also acts as an independent consultant.