Abstract— The objective of this research is to develop a learning model for programming based laboratory courses where cognitive enhancement is succeeded through active evaluation of programming assignments among peers. The key feature of the proposed model is the application of blended (both online and in-class) peer evaluation and student collaboration in an effort to improve the students’ software development skills. The overall goal of the model’s underlining pedagogy is to enhance students’ cognitive abilities, their assessment aptitudes as well as their socialization skills. The developed model has been tested in a real class context, by students who attend undergraduate laboratory programming courses. The results obtained are very significant; students expressed their satisfaction with peer evaluation activities and especially with collaborative evaluation in class, by means of a questionnaire delivered to them at the end of the semester. The collected data demonstrate that the proposed learning model is fully acceptable. Students asserted that the collaborative and peer evaluation activities benefited their knowledge and skills acquisition providing, at the same time, a pleasant educational experience. The evaluation data have also been analysed revealing interested results concerning the ability of the participating students to accurately measure the programming capabilities of their peers and support their further learning through inline annotations.

Index Terms— e-learning, peer evaluation, self-evaluation, collaborative programming testing, higher education.

I. INTRODUCTION

Within recent years, advances in computer and network technology have inspired educators to introduce the use of innovative instructional methods in Higher Education (HE), encompassing constructivist and active learning approaches which enhance learning experiences. On this basis, educators seek ways to design and offer enriched learning activities, integrating collaborative assessment activities into the learning process as an important constituent which targets learners’ cognitive, behavioral and social advancement. In parallel, self- and peer assessment enables the instructor to move from an assessment role to a role that emphasizes managing the course and teaching course concepts [1].

Collaborative performance assessment and peer and self evaluation are known evaluation methods that have been used in instruction and gained attention of researchers. Their purpose is to foster a learning approach to assessment and to develop critical thinking, autonomy and social skills. They involve the student, their peers and instructor in thoughtful and critical examination of each student’s course work [2], [3]. Concerning domain knowledge, studies have shown that it is difficult to measure how collaborative and peer reviews increase student knowledge about a specific topic due to the number of variables that impact student knowledge and course performance [4].

Peer evaluation is described as a method in which learners evaluate and comment on each other’s (as well as on their own) work according to pre-set criteria [5], [6]. Peer evaluation may be applied both as an individual and a collaborative task, either within a face to face or a technology supported environment.

Researchers and educators indicate several important advantages of peer evaluation as an instructional method [7]:

- Students get more involved in the learning process and, motivation of students for learning is remarkably increased.
- Students learn to respect each other’s opinion and become more open to criticism.
- Evaluation becomes a part of learning: students learn from their mistakes and/or success from other students; hence mistakes are not considered as failure but as opportunity for re-learning.
- Students take the responsibility of their own learning: they gradually learn to revise their work more thoroughly and make necessary adaptations after receiving their peer’s feedback.
Using peer evaluation as a self-evaluation form boosts self-reflection.

Peer evaluation enhances competition and stimulates cooperative learning.

Peer evaluation comprises the integration of several skills, putting into practice the skills for knowledge transfer.

Peer evaluation enables deep-learning instead of superficial learning.

Identified limitations for applying peer evaluation are practically due to lack of ability and maturity of students for evaluation, negative attitude towards peer education, lack of trust among peers and consideration of related activities as an additional burden \[6\], \[8\]. To overcome barriers towards a wider application of peer evaluation, transparent, well-designed, consistent criteria and process, educator support for building self-confidence and trust among peers as well as effective orchestration of group dynamics are key prerequisites for success.

This study proposes a collaborative peer evaluation model for HE with emphasis on experiments realized and obtained results, revealing the impact of the method in programming-based laboratory courses. More specifically, the paper is structured as follows: Section 2 provides a thorough description of the model and its deployment within the educational process. Section 3 illustrates the experiment conducted for the application of the model in the last two spring semesters to an Artificial Intelligence (AI) laboratory course at the Department of Informatics of the Technological Educational Institute (TEI) of Athens, and discusses related findings. Section 4 presents our plans for orchestration of further experiments to serve as the basis for a larger scale analysis and refinement of the proposed model. Finally, section 5 discusses conclusions and future work.

II. PROPOSED MODEL

Our model is a student-centered model where learning is also achieved through interactions that happen among peers. The overall goal of the model’s underlining pedagogy is to enhance students’ abilities to learn from their errors, as indicated by their peers, and from their peers’ achievements, as well as their socialization skills.

The learning process is broken down in three successive phases, as shown in Fig. 1, although there can be more or longer phases, depending on the duration of the programming-based lab course. All phases follow a general pattern, where three (at least) weeks of cognitive process are followed by an evaluation session.

An evaluation session may have the following variants: learners’ training on peer evaluation, collaborative in-class evaluation, and distance peer and self evaluation. A detailed description of the different evaluation forms used in the model is provided below.

Figure 1. The model’s 3-phase learning process

A. Training on Peer Evaluation

Early in the first phase of the proposed model, the students, in parallel to the actual lab work, are introduced to the details (guidelines and criteria) of the evaluation process and at the end of the phase each student is assigned with the assessment of two representative model assignments solving a given problem. The students’ task is to evaluate the two assignments and submit their evaluation as well as their own version of assignment for the solution of the given problem.

These model assignments are typically prepared by the instructors by strategically placing errors in the software code and in the solution’s concepts. A typical pair of model assignments contains an assignment with a well-designed and documented solution but with syntax errors and poor programming language use and the second has good and errorless code but the solution’s concepts are deficiently faced and the documentation and testing are incomplete. It should be noted that this is a specially designed training session in order to prepare students for the subsequent evaluation sessions.

Instructors have to compare submitted by students evaluation tasks against their own judgment and the results will provide them with substantial information on each student’s knowledge and ability. This insight will serve as the basis for the formation of collaborating teams in the subsequent step.

During this session, the students may work alone at home and they are provided with a questionnaire that guides them to locate the errors and perform the review and evaluation task in an efficient and objective manner. The errors contained in the model assignments are discussed in the class at the end of the first phase and the instructors point out the mistakes.

B. Collaborative Peer Evaluation

The second phase again starts with three weeks of lab courses and a new evaluation assignment.
During the second phase of the learning process, the evaluation activity takes place in the class, where students form teams and get to evaluate their own work in collaboration; each team, sitting in front of a workstation is encouraged to locate as many mistakes as possible (if any) in any of its members’ assignment. During this activity the instructors should walk around the lab, listening to the dialogs exchanged between the team members, and providing tips and guidance to help them in their task. This work may last for an hour, but it could take a whole lab’s time.

Detailed guidelines on the collaboration among students should be available to the students based on the “cooperating to cooperate” paradigm and the expected outcomes [9]. The teams, however, should also make their own decisions concerning who will be the team leader, decide about the evaluation priorities like whose work must be first evaluated and what tests are needed.

For each team, a team report is required where the team leader enters recognition for each member’s participation and evaluation outcomes. Grading of individual team projects can be included if asked by the instructor after providing a specific rubric.

According to the plan, responsibility should be shifted from the instructor to the students themselves, which take a more active role in their own learning process. Each student’s knowledge formulation is based upon self and peers’ experience [10] according to the principles of distributed constructive learning where knowledge can derive from doing, rather than receiving [11]. The team members promote each other’s learning as they verbalize and explain their solutions to the problem and decide on what needs to be done for improvement.

The teams are formed by the instructor and consist of at least two students each. Each team should optimally contain one student with higher demonstrated abilities (according to the work produced up to the course’s point) and of at least another student who appears to have a weak performance. This configuration seems to promote learning for both partners. The weaker one will definitely learn more by working with a person of higher ability and experience. This relationship also works to the benefit of the stronger partner, since operating in the position of an instructor can always make a learner contemplate more about his own knowledge.

C. On-line Peer and Self-evaluation

In the third and final phase of the evaluation process, the instructors complete the syllabus of the lab course and assign the final work to the students for on-line peer evaluation. This time there are no teams and students are asked to work individually. More specifically, they are required to work on-line, asynchronously, without the immediate guidance of the instructors, using any available communication tools. When this step is completed the students are asked to evaluate their own assignment’s results under the assumption that, by then, they will have acquired sufficient experience so as to carry out this process successfully.

The underlined pedagogy of peer and self-evaluation of this phase is to support students in enhancing their knowledge skills by asking them to actively compare their peers’ assignments with the instructor’s predefined rubric. Such a rubric must be simple in understanding, detailed and fair and is usually presented in the form of a questionnaire format.

On one hand, peer evaluation is the most efficient way of helping students receive scores and feedback in short time, especially in courses with high enrollment numbers. On the other hand, peer evaluation, following self-evaluation, supports students’ self-reflection on their achievements and the level of their own performance.

III. APPLYING THE MODEL

A. The AI Course Experiment

The course of AI is taught in the 6th semester of studies at the Department of Informatics of the TEI of Athens. It is an obligatory course comprising 2 hours of direct instruction (lectures on AI theory) and 2 hours of laboratory training. The students, during the lab part of the course, have to solve a specific problem developing search algorithms in common lisp language and expert systems in clips environment, two programming environments totally new to them. They are split into 4 different laboratory groups so that a maximum of 20 students participate in each group. Every week a programming assignment is assigned to the students, who are given a week’s time to carry out and deliver their work, either working alone or in pairs.

During the two last spring semesters, we tried to base our instructional pedagogy on the aforementioned collaborative learning model. The results presented in this paper concern the spring semester of 2013.

The training phase was easily designed following exactly the demands of the model.

Our approach on collaborative evaluation in the second phase was to form teams of two to three students with different levels of programming skills to cooperate in class in a synchronous way, in order to find and fix at first errors at the assignment of the weaker team member. Then, the rest of the assignments were checked with the contribution of all team members. At the end of the evaluation process, the team members were asked to write a report on a predefined evaluation sheet as shown in Table I and deliver it to their instructor, handing in any new version of code that had resulted from the collaboration effort. Throughout this experiment 35 teams were formed and 50 assignments were corrected and re-submitted followed by their evaluation sheet.
TABLE I. COLLABORATIVE EVALUATION QUESTIONS

| 1. Name of the student’s assignment that is evaluated |
| 2. Names of the students that are collaborating in evaluation |
| 3. General comments about the program to improve (correctness, completeness, quality etc.) |
| 4. Code of the functions that need improvement and reason for the improvement Comments on the test cases and improvements performed |
| 5. Comments on the test cases and improvements performed |
| 6. Comments about the documentation of the assignment |
| 7. Comments of student who is evaluated about his knowledge enhancement through the evaluation process |
| 8. Comments of students who are evaluating about their knowledge enhancement through the evaluation process |

During the third phase, collaboration was realized by means of the communication tools provided through the department’s e-learning platform. Each student received for evaluation two anonymous assignments of classmates belonging to different lab groups. A specially designed evaluation guide, helped students through the tasks of assessing different aspects of the assignments like the correctness, completeness, effectiveness, and quality of the code, the completeness of test cases for the verification/validation and the quality of the documentation of the programming assignment. For each different aspect, students were asked to provide comments and give their own mark as well as a final mark for the complete assignment. For completing the task, students were asked to comment on their own submitted work following the same criteria as those used for their peer’s evaluation. The reason of anonymia at this phase is to ensure that an unbiased evaluation would take place. For socialization purposes, the names of authors were announced after the evaluation.

After the deadline of the peer evaluation activity the instructor’s marks on assignments were announced so that the students could assess their own judgment against that of their instructor.

B. Collection of data, analysis and findings

The training phase was very successful as students were experiencing for the first time a similar learning activity. They felt very comfortable not having to write their own assignment from scratch but instead having to evaluate two already existing assignments, even if they had to discover where errors occurred. The uploaded students’ assignments at the end of the activity were of good quality and instructors’ feedback was an easy task.

The benefits of the approach followed during the second evaluation phase were twofold according to students’ comments included in the evaluation sheets. Weak students gained help from a colleague in order to complete their assignment and better understand the problems they had faced, thus enhancing their knowledge level. On the other hand, students with error-free assignments had a good opportunity to “teach” their colleagues and, hence, also improve their knowledge level, understanding certain aspects of the program that were fuzzy until then. But most importantly, they had improved their skills, having to locate the problems in peers’ programs, a skill which is essential in mastering a programming language.

Another benefit was that classmates had the opportunity to know each other, establish a friendly relation and support each other throughout the semester.

During the third evaluation phase, out of the 71 students who had uploaded their assignments to the e-learning platform and were subsequently asked to participate in the peer evaluation activity, getting a bonus just for their participation, only 9% were not consistent.

The participating students have delivered 109 evaluations as some of them haven’t evaluated both received assignments. Most of the students have adequately considered their classmates’ work commenting on it and justifying their marks.

The percentage of evaluators who succeeded to give exactly the same mark as their tutors’ were 49%, most of which concerned evaluations of assignments of very high or very low quality (38%). An additional percentage of 20% of evaluators gave a close mark (a difference of minus one to plus one) to their tutors’ one, thus augmenting the number of successful evaluations to 69%. From the rest evaluations, 23% were too strictly and 8% too leniently marked.

Regarding the analysis of gathered data, a univariate analysis has been used to analyze the collected data. All the following associations were assessed with the $x^2$ test.

The only interesting finding from this analysis was that the weaker students who evaluated stronger ones answered that everything was correct about the program (which was the case), but also answered that the assignment included all the required testing cases, which was not true for almost half of the assignments.

At the end of the semester an anonymous survey was conducted to assess the effectiveness of the model through the students’ opinions. Overall, we got back 31 answers out of a total of 85 students who attended continuously the course and exactly the half (62) of those who have successfully completed it.

Questions were grouped into five sections. The first section consisted of various general questions concerning the lab group the students participated in, if they were attending the theoretical part of the course, etc. The second section comprised questions related to the satisfaction of the students on the organization of the lab, the connection with the theoretical part of the course and the overall satisfaction. The third part included questions about the quality of the educational procedure the teacher followed. In the fourth section, questions regarded the collaborative and peer evaluation processes and the degree of contribution of the model to the students’ cognitive enhancement. The final section involved open-ended questions, were students commented on the best and worst aspects of the laboratory course. Most questions concerning
satisfaction level of the students were on a 5-point Likert-scale (1 - strongly agree, 2 - agree, 3 - neutral, 4 - disagree and 5 - totally disagree).

On this basis, the following outcomes were recorded: All students declared that they have succeeded to the course and a percentage of 45% of them declared that they got a final mark over 8.0. Comparing it to the fact that in our tables only the 30% of students got a mark over 8.0, we can conclude that high marked students had volunteered more to answer the questionnaire. They believe, in a 75%, that the allocated assignments through the course helped them to better understand the theoretical part of the lesson, whereas only 5% stated the opposite. Lastly, 72.5% expressed that they are happy (only 2.5% dissatisfied) about the grading criteria.

Students were very satisfied with the way the educational material was presented, they were pleased that they could easily ask questions and they were excited that the instructors stimulated their interest for the course.

Almost all of them declared to be good or very good in programming (97%), 74% of them were very or fully satisfied with the final mark they have got and only 9% claimed to disagree with it. Concerning their participation to the third phase of peer and self evaluation, 52% declared that they participated with pure interest, 25% because of the bonus mark they would get and 23% from curiosity for their peers’ work. Most of them (94%) found the peer evaluation an interesting activity that had beaten the class routine.

Some of the most interesting results of the questionnaire concerning the third phase of the model’s implementation are presented in the following tables.

TABLE II. QUESTIONNAIRE ON THE EVALUATION PROCEDURE

<table>
<thead>
<tr>
<th>Questions</th>
<th>NOT ENOUGH</th>
<th>ENOUGH</th>
<th>MUCH</th>
<th>A LOT</th>
</tr>
</thead>
<tbody>
<tr>
<td>How much effort have you put to this activity?</td>
<td>3%</td>
<td>16%</td>
<td>29%</td>
<td>52%</td>
</tr>
<tr>
<td>How much have you tried to make helpful comments?</td>
<td>16%</td>
<td>23%</td>
<td>48%</td>
<td>13%</td>
</tr>
<tr>
<td>How helpful were the comments from your evaluators?</td>
<td>24%</td>
<td>24%</td>
<td>44%</td>
<td>8%</td>
</tr>
<tr>
<td>How similar were the two evaluations you have received?</td>
<td>12%</td>
<td>36%</td>
<td>48%</td>
<td>4%</td>
</tr>
<tr>
<td>How representative to your assignment level is the higher mark your evaluators gave you?</td>
<td>7%</td>
<td>37%</td>
<td>37%</td>
<td>19%</td>
</tr>
</tbody>
</table>

From the analysis of the collected data we can conclude that although most students declare that they put a big effort to this activity, not all of them believe that they tried hard to really contribute in the knowledge enhancement of their peers.

Even worse is their own experience of the evaluations of their own assignments concerning the comments they have received. This is well explained if we take under consideration that the half of those who replied to the questionnaire was high marked students.

Most of the students declare that both evaluations they had received were similar (84%) and that the higher marking between them was representative to their assignment’s level (74%).

TABLE III. QUESTIONS FOR EVALUATION ENHANCEMENT

<table>
<thead>
<tr>
<th>Questions</th>
<th>NO</th>
<th>YES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Would you try harder in this activity if the quality of your evaluations were also evaluated and marked?</td>
<td>42%</td>
<td>58%</td>
</tr>
<tr>
<td>If the evaluations were not anonymous, would that affect the effort you made for this activity (including marking)?</td>
<td>63%</td>
<td>37%</td>
</tr>
</tbody>
</table>

As it can be concluded from students’ answers, opinions are divided between whether the evaluations’ assessment and marking by the tutors would affect the students’ effort for providing feedback of better quality. Concerning anonymity, those who don’t care if evaluations are anonymous or not, are almost double from those who think that if they had to provide an eponymous evaluation this would affect their effort.

In the last questionnaire section we quote some representative comments: “We appreciate the pioneer way we helped each other to conclude in a correct and complete assignment”, “Evaluations should not be anonymous, we accept responsibilities for what we write”, “I did not like the collaborative procedure, because it depends on whether your colleague is in a position and in the mood to help you”.

IV. PLANNED EXPERIMENTS

The proposed collaborative peer evaluation learning model fits very well to the agile methodologies of software engineering, aiming to produce software in a rapid and flexible manner. Agile programming also uses the concepts of small team collaboration and continuous iteration of the programming lifecycle, where work is broken down into repetitive cycles of design, implementation and testing phases.

For the forthcoming semester we are planning to apply elements of agile software development to the already proposed learning model, in an effort to enhance the
learning process in the context of an Object-Oriented programming lab course. The employment of agile methods is considered very suitable for core programming courses. The most significant change will be in the nature and duration of the final project. Instead of allocating only the last month for the implementation of the project, the project will now be in constant implementation mode for the whole duration of the semester. According to the agile programming concepts, the project will be broken into three increments of work, each lasting for 3 weeks. Moreover, instead of working alone, students will form pairs and will work on the project together as a team.

An additional experiment will be applied next semester in Compilers course. This is not a typical programming course but still students have to experiment in compilers’ design through programming. The lab part of the course is divided in three parts and this is very convenient for the implementation of our learning model. In the first part, which will be used for evaluation training, the students have to design the regular expression and automata for the grammar of an exemplar language and use them to write the code of the language’s lexical analyzer in C/C++. In the second phase the students have to develop the lexical analyzer using Flex. During this stage, the collaborative evaluation will be carried out. In the third part, the students must develop the syntax and semantic analyzers using Bison and prepare a final assignment containing the outcomes of all lab activities. These assignments will be used for peer and self evaluation at the end of the semester.

V. CONCLUDING REMARKS

The paper presents a collaborative learning model based on the peer evaluation paradigm that aims to enhance the learning process in the context of programming-based laboratory classes. The key feature of the proposed model is the application of peer evaluation and student collaboration (both online and in class) in an effort to improve the students’ software development skills.

The learning process is broken down into three successive phases and each phase consists of cognitive work in the lab followed by different forms of collaborative evaluation.

The concepts of the learning model were applied in the context of an Artificial Intelligence laboratory course and the effects of the process were studied with the use of especially designed questionnaires. The results have been very promising, revealing that the proposed learning model benefited the students’ knowledge and skills acquisition and their collaboration capabilities, providing them also with a pleasant educational experience.

REFERENCES


