CLASSIFYING ROCKS: A PRACTICAL EXERCISE TO LEARN PROGRAMMING THROUGH GEOLOGICAL CONCEPTS

A. Alonso, A. Casillas, L. Ortega, M. Penagarikano

University of Basque Country UPV/EHU (SPAIN)

Abstract

The Undergraduate Degree in Geology of the University of the Basque Country contains an “Introduction to Computation” subject that is taught in the first semester of the first course. Students are generally very little interested in this matter, mainly because they think that programming skills are no related to any geological subject they are interested in.

In this paper, we discuss the implementation of a new activity in the context of the subject “Introduction to Computation” and focused on increasing the motivation of the students on the programming subject. The activity consists on the development of an exercise which combines programming skills and real geological problems, and it is supervised by a multidisciplinary team of teachers from different knowledge areas: teachers from the subject “Introduction to Computation” (henceforth computing teacher) and teachers from the subject “Complements of Geology” (henceforth geology teacher).

The proposed exercise consists of three phases. First, the geology teacher introduces a real geology-related problem and explains how it could be resolved. On a second phase, the computing teacher explains how to automatize part of the resolution process in terms of a programming language. On a third phase, and once the students have implemented the algorithmic solution to the problem, they solve real examples of the proposed geological problem using the previously created computing tool. On this last phase, both the geology and computing teachers help the students with any geological or programming related question.

The selected geological problem was the Classification of Rocks. The students are presented a classification method based on binary questions (two possible answers, YES/NO) which can be implemented in terms of complex/nested If-Then-Else statements. The exercise introduces some new geological concepts and serves as a test-bed for the programming skills, and the students create a useful computer program that will be finally used to classify a collection of rocks.

Some interviews carried out after this experience showed that the students were satisfied with the exercise and that they had gained interest in the programming skills.

Keywords: programming, geology, rocks classification.

1 INTRODUCTION

Students of the Undergraduate Degree in Geology of the University of the Basque Country take an introductory subject to computing. The students show a lack on basic geology knowledge and they have no prior experience with the fundamentals of programming languages. Moreover, the students do not understand the need to learn any programming language, since they do not perceive the computation as an important skill for Earth-related sciences and they do not identify any application of the programming skills to the subjects of their discipline. The assessment results obtained over the last courses reflect problems on the assimilation and understanding of the basic programming concepts, as well as a high percentage of unmotivated students.

The motivation of the students and their learning process are univocally related [1], even more on the first course, which includes laboratory and field practices, and is critical for the development of their interest on the chosen career [2]. The design of learning activities that are based on topics relevant to the students’ curiosity can promote their intrinsic motivation [3]. Such activities could include strategies such as using local examples, teaching with news events, using technology, etc. The motivation of the students may also increase when they feel part of the learning process, whereas it will decrease when they do not perceive any element of autonomy. On the other hand, the collaboration among students in the classroom promotes their interest in the subject matter and the construction of the knowledge with higher level of achievement [4].
In this paper we propose an experience that includes the modification of one of the three lab exercises scheduled in the subject. New activities have been designed based on the previous teachers’ lab experiences. Those activities aim to consider the students as valued members of a learning community. Several specific and key transversals competences will be worked: basic knowledge on geology and technology, analysis of the problems, application of the knowledge and the programming techniques, understanding the formulation of the problem, using an actual programming language and working as a cooperative and creative team. With the proposed exercise we are giving to the students the opportunity to work with a real geology-related problem that will be solved using programming tools.

The rest of the paper is organized as follows. Section 2 analyzes the current state of the subject prior to the implementation of the new lab exercises and presents the quality indicators that will be used to measure the improvement of the students. Section 3 introduces the new lab activities designed to motivate the students. Section 4 analyzes the results in terms of the quality indicators and the paper ends with the discussion of Section 5.

2 BACKGROUND AND QUALITY INDICATORS

The subject “Introduction to Computation” has been taught for the last seven years, since the school year 2010/2011. The main objective of the current work is to increase the attention and interest of the students in the subject. A direct way to monitor such progress is to compare the amount of students that leave (do not attend to the final exam) and approve the subject before and after the proposed changes are implemented.

Table 1 shows the percentage of students who did not attend to the exam (were not evaluated) and the students who did attend during the last four school years (before the current 2016-2017 school year). Taken into account the students who were evaluated, Table 1 shows the percentage of students who passed the subject and the percentage of students who did not.

<table>
<thead>
<tr>
<th>School year</th>
<th>Not Evaluated %</th>
<th>Evaluated</th>
<th>Pass %</th>
<th>Fail %</th>
</tr>
</thead>
<tbody>
<tr>
<td>2015-2016</td>
<td>33</td>
<td>25</td>
<td>75</td>
<td>25</td>
</tr>
<tr>
<td>2014-2015</td>
<td>33</td>
<td>25</td>
<td>75</td>
<td>25</td>
</tr>
<tr>
<td>2013-2014</td>
<td>29</td>
<td>46</td>
<td>54</td>
<td>46</td>
</tr>
<tr>
<td>2012-2013</td>
<td>42</td>
<td>50</td>
<td>50</td>
<td>50</td>
</tr>
</tbody>
</table>

In [5] we concluded that “Introduction to Computation” is not a subject of interest for many students. This low interest in the subject could explain the high number of students that usually leaves the subject and does not attend to the final exam as we can see in Table 1. The lack of students’ motivation led us to think that these students do need a new teaching approach in order to awaken their interest in the matter. In this way we decided to rewrite the lab exercises. Next section presents one of such exercises: rocks classification.

3 ROCK CLASSIFICATION: A PRACTICAL GEOLOGICAL EXERCISE

The classification of rocks is a fundamental geological task. Rocks are key elements in lectures, laboratory and field practices, but often it is a hard exercise for the students. Although the lecture classes include real samples based rock identification exercises, the students are not able to apply these skills later on in the laboratory or the field practices. Geology is an observational science: it is essential to know how to describe the materials and classify that information into categories. The
classification of rocks is an exercise of observation and promotes the development of clustering and labeling criteria.

To solve this problem, we propose to the students to create a computer program that will implement *The Rock Identification Key* [7], a flow diagram with different aspects that the students must examine in the rock, following a collection of binary questions (see Fig. 1). The mineralogical composition and the texture are the key features that they must observe in order to classify the rocks. The textural criteria are considered first, because they provide the best signature for rock origin and allow classifying into the three great genetic categories: sedimentary, igneous and metamorphic. The texture is related with how the minerals are resembled together: the size, shape and the arrangement of the grains (for sedimentary rocks) or crystals (for igneous and metamorphic rocks). The homogeneity or the uniformity of composition throughout is also observed. The mineral identification and the quantity of each mineral in the rocks are also used for classification. Sometimes, the minerals are too small to be identified and then, the color of the rock is used. Other times, there are materials that are classified on the basis of their taste.

At the beginning of the exercise, the teacher of the subject “Complements of Geology” tries to convey to the students how important the ability to classify rocks is for a geologist. Students are shown rock samples as those of Fig. 2, and the teacher explains the terminology that they will find in the *Rock Key* questionnaire.

Once the rock classification problem has been fully introduced from a geology point of view, the teacher of the subject “Introduction to Computation” takes over and analyzes the problem and the proposed solution from a programming point of view. The students must create a program that will implement the solution based on the binary questions pattern of the *Rock Key*. When the user runs the program, he/she will be asked questions, and the successive questions will vary depending on each answer, until arriving at a solution (the class of the rock has been guessed).

When the students finish the programming task, they are given the rock samples and the must use the created program to classify those rocks. During this test phase, geology related questions (typically, they don’t fully understand a question) or programming related ones (typically, the program crashes) are answered by both teachers. If they don’t get any problem but the classification result is wrong, they are helped to detect if it is an error due to a wrong answer or on the contrary it is a programming bug.

From the programming point of view, the resolution of the proposed problem can be implemented using nested *IF-THEN-ELSE* statements and serves as an example of the use of conditional flow-control structures. Moreover, this seemingly simple problem can be used to show to the students more sophisticated implementations, based on recursive functions or finite state automata.
4 RESULTS

Our main objective is to increase the attention and interest of the students in the subject. A direct way to monitor such progress is to compare the amount of students that leave (do not attend to the final exam) and approve the subject before and after the proposed changes are implemented. The amount of students that leaved and approved the subject before the proposed changes have been showed in Table 1. Table 2 shows the same numbers for the 2016-2017 school year, the first school year in which we have put in practice our proposal.

Table 2. Percentage of students who did not attend to the exam (were not evaluated) and percentage of pass and fail students (among those presented) during the 2016-2017 school year, for each language related student group: Spanish and Basque.

<table>
<thead>
<tr>
<th>Student Group</th>
<th>Not Evaluated %</th>
<th>Evaluated</th>
<th>Pass %</th>
<th>Fail %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spanish</td>
<td>42</td>
<td>59</td>
<td>41</td>
<td></td>
</tr>
<tr>
<td>Basque</td>
<td>27</td>
<td>57</td>
<td>43</td>
<td></td>
</tr>
</tbody>
</table>

In order to complement the traditional satisfaction inquiry, we have made another questionnaire that include questions more focused on the subject and open questions, so that the students can express directly what they think about the new changes introduced in the programming subject.

The questions focused on the subject that we have included are:

1. The subject has been focused on the use of programming techniques to solve real situations in the field of Geology.
2. The programming of the subject takes into account the development of competences proper to our future profession.
3. The instructions given to program during the laboratory practices are clear.
Regarding the open questions, the objective of this type of questions was to know the students’ suggestions. The main conclusions that we have get are two: 1) They do not have enough training on concepts of geology; 2) They prefer geology based exercises than other more general exercises.

We have done personal interviews with three students of the subject in Spanish all of them first-time students. We invited to the interview to an student who did not approve the matter in the two previous courses, but he did no attend to the interview; we must notice that this school year the student has passed the matter. After the interview with the students we conclude that they are agree with this new type of exercises and they asked us to convert all the lab exercises into problems based on geological problems, even more they proposed us to do the same transformation with the exercises showed in the seminary classes. The students understand better the algorithmic structures when the exercises solved are based on geological problems.

In the previous article [5] we mentioned that in the 2016-2017 school year we were going to pass to the students the traditional satisfaction survey. We have passed it, but the results are not available yet.

5 DISCUSSION

This paper has presented a collection of actions designed to increase the attention and interest of the Geology grade’s students in the subject titled “Introduction to computation”. We believe that addressing some real activities, more related to their future work as geologist, could change the perception that students have of the usefulness of programming skills.

In particular, we present a new lab exercise designed to learn programming skills through the implementation of a specific solution to a real geology related problem: the rock classification. The proposed resolution, The Rock Key, can be implemented using nested IF-THEN-ELSE statements, and allows developing more sophisticated implementations that can be further analyzed in the seminars.

From the programming point of view, the resolution of the proposed problem can be implemented using nested IF-THEN-ELSE statements and serves as an example of the use of conditional flow-control structures. Moreover, this seemingly simple problem can be used to show to the students more sophisticated implementations, based on recursive functions or finite state automata.

ACKNOWLEDGEMENTS

This Innovative Project Research project has been achieve with the financial support of the UPV/EHU (SAE/HELAZ).

REFERENCES


