EasyCoding – Methodology to Support Programming Learning

Marcela Viana P. Almeida
Research Centre in Digitalization and Intelligent Robotics (CeDRI),
Polytechnic Institute of Bragança, Portugal
a42929@alunos.ipb.pt

Luís M. Alves1
Research Centre in Digitalization and Intelligent Robotics (CeDRI),
Polytechnic Institute of Bragança, Portugal
lalves@ipb.pt

Maria João Varanda Pereira
Research Centre in Digitalization and Intelligent Robotics (CeDRI),
Polytechnic Institute of Bragança, Portugal
mjoao@ipb.pt

Glívia Angélica R. Barbosa
CEFET Minas Gerais, Brasil
glivia@cefetmg.br

Abstract
Knowing that the programming curricular units in the first year of engineering courses have a high failure rate and, assuming that this failure is due, in large part, to the lack of motivation and the lack of autonomy of the student to program in context outside the classroom, a methodology based on activity guides using attractive web platforms is proposed. The proposed methodology aims to facilitate both the planning of activities by the teachers and the autonomy and motivation by students. In order to receive a first feedback about this work, the methodology is being used by programming professors from Polytechnic Institute of Bragança, but in the near future it will be also evaluated by professors from the Federal Center of Technological Education of Minas Gerais and from the Federal Technological University of Paraná, both from Brazil. Following this work, a system is being developed that allows the automatic construction of guides based on exercises available from the web and systems that facilitate the collection of solutions and analysis of results.

2012 ACM Subject Classification
Human-centered computing → Visualization systems and tools;
Software and its engineering → Imperative languages; Social and professional topics → Computer science education

Keywords and phrases learning programming, teaching programming, automatic activity guides, programming motivation

Digital Object Identifier 10.4230/OASIcs.ICPEC.2020.1

Funding This work has been supported by FCT – Fundação para a Ciência e Tecnologia within the Project Scope: UIDB/05757/2020.

1 Introduction

School dropout and failure in higher education has been the subject of many research studies. Student dropout is an international problem that has social consequences and affect the results of educational institutions [1]. This is due to several factors, such as the lack of

---

1 Corresponding author
motivation and dedication of students in the face of the challenges that are proposed to
them inside and outside the classroom. Following this context, Mendes [14] quotes that the
first programming curricular units at the Department of Informatics Engineering from the
University of Coimbra suffer from high failure and dropout rates, as reported in many other
high educational institutions. As Silva Filho [18] quotes, in Brazil, the private sector invests
around 2% to 6% of higher education institutions’ revenues in marketing, in order to attract
new students.

However, in order to keep students enrolled, just short and punctual experiments have
been made in higher education contexts [17, 13, 16]. Moreover, developing a programming
logic can be a complex and challenging task. This is because, according to Moreno [15]
and França et al. [8], in addition to knowing the basic instructions, the individual needs to
understand how to use them to solve different problems. Furthermore, the applicability of this
content in other areas has expanded the profiles of people interested in learning it (e.g., high
school students). Therefore, it is necessary to diversify the approaches to teaching/learning
programming so that they are suitable for different profiles of interest [8, 15]. Regarding
this issue, given the constant unsatisfactory results in the first programming disciplines of
technology courses in Portugal and Brazil, a larger study on new methodologies to support
programming learning become essential in the search for better teaching quality and greater
enrollment of students.

Following the results analysis of the project NoviBraga [4], the objective of this study is
to avoid the lack of dedication of the first year programming students. To this end, teaching
methodologies are being developed to improve the quantity and the quality of the slots of
time that the student uses to program outside and inside classroom and consequently improve
the learning outcomes. According to Gomes and Mendes [9], when talking with programming
teachers, most of them claim that students don’t know how to program because they don’t
know how to solve problems and don’t have enough mathematical background. There are lots
of tools on the web that can be used by programming students and almost all of them have
some motivating features like animations, control flow graphs, interactive debugging, syntax-
direct editors, collaborative programming, chats, programs to complete, games, submission
platforms, rankings and so on. All these features can turn the programming activity more
attractive and easier if they are correctly explored. Moreover, these tools can be combined
to join the best of each one. In order to achieve this, teachers also need some support to be
able to propose the appropriate set of activities along the semester and get feedback from
the students. An activity guide is needed, and the exercises must be carefully chosen. It is
difficult to find in the web these activity guides already constructed. The main contribution
of this work is to propose a methodology based on web platform activities and to create
a system to automatically produce activity guides. These guides are based on free web
platforms whose features we believe that can motivate students to program. This work
presents two research questions:

1. Does the use of web systems that allow the resolution of programming exercises motivate
students to work outside the classroom? (Section 3)
2. Is it possible to systematize and automate the creation of programming activity guides
for both the teacher and the student? (Section 4)

Besides this introduction where some related works are presented, the section 2 introduces
the methodology proposed. The application of the activity guides is discussed in section 3
and based on this experience, a system to automatically generate those guides is presented in
section 4. In section 5, some conclusions will be presented as also some clues for future work.
2 Methodology

In order to follow our approach, the following steps are proposed:

1. Collection of information on C programming web teaching platforms;
2. Creation of a classification system for web platforms, with the objective of using them inside and outside the classroom context;
3. Classification of platforms found and creation of a website to make this study available to C programming teachers;
4. Creation of activity guides to be developed during the first programming semesters with the students, with respective monitoring;
5. Validation of the methodology, based on the results of the monitoring, the students’ academic success and questionnaires for collecting opinions from teachers and students;
6. Development of a system that allows teachers to assemble and generate new activity guides, according to characteristics pre-defined by the classification model.

The result of the three first items can be seen in [3]. At the end of the project’s development, the results will be analyzed and validated, and future perspectives and proposals for improvement of the study will be pointed out.

3 Application of the Activity Guides

The present study considered an introductory programming course (Programming I) in the context of higher education at Polytechnic Institute of Bragança (IPB). This course unit has 218 enrolled students spread over two courses, namely, Informatics Engineering and Management Informatics. Programming I is in the first year, first semester of our two courses. At the end of these course unit, the student is expected to be able to design solutions and implement C programs that solve small/medium complexity problems. In order to do that, the student must apply concepts of imperative programming in the C programming language, coding function-based structured programs to manipulate data structures. The student must also be able to use an Integrated Development Environment (IDE) including the debugging tool to get successful solutions. In this case, the IDE is the Microsoft Visual Studio Community 2019. The syllabus of this course contains C language topics that are commonly taught in introductory programming at the university level. Like other cases found in the literature, Programming I also reveals a low student success rate. The teaching-learning programming process is a difficult task, as is known to the community involved. This process faces several challenges, the most important of them, it is to improve student motivation to learn and to ensure that they do not give up learning. Another important challenge is to enroll the students to work outside of the classroom. Aware of these challenges, in the first semester of academic year 2019/2020 we applied the proposed approach to hold the students involved in programming tasks. Thus, we create a set of activity guides that cover the whole syllabus of programming I. These guides use a set of web platforms with exercises well contextualized. In the classes, the teacher gave a brief description of each of these guides, whose exercises were solved by the students inside and outside of the classroom. In a first phase, we proposed the activity guides to 65 students that are divided into two different classes. We decided to involve only these students for two reasons:

(i) The teacher, one of the authors of this paper, taught both classes, so he had full control of the students;
(ii) The idea is to start just as a pilot project and adjust the guides during the semester.
The Figure 1 presents the results of the activity guides application. We enumerated the activity guide from 1 to 5 and they are divided by topic. For each topic we have more than one activity guide. The table shows also the web platforms used by the students to do the tasks. A qualitative research was carried out on the main tools available online, which support teaching C programming, focusing mainly on features considered useful for combating school dropout. The chosen platforms are URI Online Judge, Codeboard, CPuzzles, C Tutor, CodinGame and Coderbyte [11, 5, 12, 10, 7, 6]. They were chosen based on their characteristics like quantity of exercises, contents scope, ease of use, free access and some extra features like tutorials, animation, collaborative work and automatic assessment. All these web platforms have different purposes and they cover a set of different functionalities:

1. URI Online Judge is a web portal that offers a variety of programming problems for the students to solve and share knowledge, besides having a teachers’ module called URI Online Judge Academic that lets the teacher manage the activities, students’ submissions and grades, without having to use any other system. Its main features include real-time correction, available problems separated by categories and use of gamification concepts, with the application of a of a badges and ranks reward system.

2. Codeboard is a web-based IDE to help the students to learn programming by allowing the teachers to create exercises as they wish and letting the students run and compile their own solutions inside the IDE and after submitting to the teacher, that has total access to the student answers. Codeboard also allows automatic grading to help the exercises correction and has the possibility of exporting the results in spreadsheet format.

3. CPuzzles is a repository that contains a collection of C puzzles and solutions to supplement the activities and assist the teacher. The puzzles inside the platform are divided in eight groups, each one regarding a C subject. Each puzzle has a difficulty rating, in order to help the teacher choose which exercise better meets students needs. Besides that, the teacher can find an example of solution and a "code skeleton", which helps the student to start the exercise.

4. C Tutor is a tool that assists teachers and students when writing code, making them understand what happens as the computer runs each line of code and visualize each state of data structures being used. When applicable, C Tutor also helps the visualization of the contents present in the heap and the pointers. Finally, the system allows collaborative work, by letting different users change the same code simultaneously and has an online chat that allows students to discuss each problem.

5. CodinGame is a gamified platform for teaching programming languages. The system uses game puzzles with high level animation and different difficulty levels. The website has its own IDE where the students can run and compile their own code and submit their result to the platform that has its own automatic correction. CodinGame also has a reward system with the application of badges and ranks. There is still a forum dedicated to the users for any doubts about the puzzles.

6. Coderbyte is a web system developed to offer challenges and courses with the aim to help users to prepare for upcoming job interviews, as well as practicing more programming. The system has an IDE where the students can run, compile and test if their solution is valid, by going through the system’s automatic correction. The code challenges range in difficulty and can be solved using different programming languages.

In Figure 1 we can see the code and the name of each exercise made by the students. These codes were obtained by the platform itself. In the last three columns of Table 1 we can see the number of total students to whom the activities were proposed, the number of student submissions and the number of successful submissions. Based on these numbers we quickly
realized that the student involvement is much greater in the first activities and it has been decreasing as the subjects have become more complex. The deadline for students to complete the tasks of the last three activity guides is still open, so it is expected that more students will complete them. The Coderbyte and CodinGame web platforms had less adherence by students. In fact, some of the students referred some difficulties to use and to understand the problem statement of these platforms. All the activity guides were transferred to the students using Sakai platform. This is a web platform similar to the Moodle used in IPB. During the classes, the teacher briefly explained the tasks for each of the activity guides. The students received the activity guides by institutional email. The text of the email always referred some encouragement words. Unfortunately, not all students read institutional email, especially those who miss classes. We need to address this issue during the next semester and use a different approach to convince students to do the activities.

After each activity the teacher opinion was collected and used to improve the next guides. The idea now is to intensify the construction of new activity guides using the platforms that had more success from the students’ point of view and also the ones that are more appropriate for the contents of Programming II. Since data structures will be better explored, we are aware that animation tools will be more adequate and useful.

### 4 A System to Automate the Activity Guides Construction

A system (Figure 2) called EasyCoding is being developed to automate the guides construction and help teachers to produce new activity guides in order to use them with the students inside and/or outside the classroom, with the aim of facilitating their planning and to motivate the students by performing these exercises. The technologies being used for the development of this platform are Javascript with JQuery library to simplify the scripts interpreted on the client-side browser, jsPDF library which is a widely used solution for client-side PDF generation, in order to generate the activity guides for teachers and students, and Bootstrap front-end library to help prototype the system for a friendly and responsive experience. As a
static platform, the system is being hosted on GitHub Pages host service. Therefore, the system is capable of automatically generating activity guides, in order to explain to students and teachers the correct use of each chosen web platform. The proposed exercises were collected from the studied web platforms (when they had exercises available as a feature) and also from the book “Linguagem C”, by Luis Damas. When using the system, the teacher only has to choose:

1. Inside or outside the classroom context.
2. Programming subject.
3. Platform in which s/he wants to apply the exercises.
4. The exercises themselves.

Each exercise will have an estimated time in which the teacher can rely on to compose her/his activity guide. Actually two guides will be produced, one for the teacher to prepare the session and one for the student, that will receive it from the teacher, to follow the guide. The results will be automatically collected, and the teacher can use them as s/he wants. The system proposed is already available in [2].

The main reason for the generation of these two types of activity guides (teacher and student guide) is that each web platform being used has its own way of use. Some of them, like URI Online Judge and Codeboard have different functionalities for teachers and students. Because of this, each guide is developed to help their understanding of the steps they have to complete to accomplish the task. Thus, the teacher’s guide will assist the teacher on how to use the web platform chosen to add the activities needed and show them to the students. The students’ guide will assist them on how to have access to the web platform, perform the given activities and send their results to the teacher. In order to evaluate the acceptance of the analyzed web platforms, an interview was carried out with the students who participated in this study. The interview revealed that the URI Online Judge, Codeboard and Python Tutor platforms had a greater acceptance among the students. According to them, the other ones seemed complex or without significant benefits. Due to this result, we opted to keep on working, this next semester, only with these 3 web platforms and CPuzzles, to collect exercises. Moreover, the EasyCoding system is being implemented to provide activity guides using these 4 platforms also. Examples of generated teacher and student’s guides are shown in Figure 3.
Figure 3 First page of a teacher and a student’s activity guide applied.

5 Conclusion and Future Work

In this paper, a methodology based on the creation of programming activity guides was described. The concrete application of the proposed approach to IPB students was explained as well and the results from this initial research were analyzed. From the first semester of application, we are able to conclude which are the most appropriate tools and what kind of activity guides should be produced to enrich our system and improve the student’s motivation. The student opinion about this was also collected and used to improve the guides. The developed system has a database of activities that will be used by the teachers to automatically generate more guides as they wish. At the end of this second semester, the success rate of the Programming I and Programming II subjects will be analyzed and this study will be able to show if the application of these activity guides with the use of web platforms that have motivating features was satisfactory. For that, student and teacher opinions will be collected again.

References