

A Comparison of Introductory Programming Courses between Portugal and Serbia

Luís Alves*, Pedro Rangel Henriques**, Vladimir Ivančević***, Maksim Lalic****,
Ivan Luković***, Maria João Varanda Pereira*, and Paula Correia Tavares****

* Instituto Politécnico de Bragança, Departamento de Informática e Comunicações, Bragança, Portugal

** Universidade do Minho, Departamento de Informática, Braga, Portugal

*** University of Novi Sad, Faculty of Technical Sciences, Novi Sad, Serbia

**** Instituto Superior de Engenharia do Porto, Departamento de Informática, Porto, Portugal
lalves@ipb.pt, pedrorangelhenriques@gmail.com, dragoman@uns.ac.rs, lamaksim@uns.ac.rs,
ivan@uns.ac.rs, mjoao@ipb.pt, pct@isep.ipp.pt

Abstract - The goal of the study was to learn more about differences in academic performance, teaching, and assessment between introductory programming courses in Portugal and Serbia. Data about a selection of higher education courses were compiled and a comparison of student performance and methods employed in the classroom was performed. The obtained results are expected to serve as a basis for subsequent research on how to improve the current teaching practice in the two countries.

I. INTRODUCTION

Basic programming education has become ubiquitous in higher education all over the world and it is now a common part of various study programmes in science, technology, engineering, and other areas. As programming is being taught to a large number of students, there is a great pressure to provide satisfactory education. This in turn is expected to facilitate wider application of information and communications technology (ICT), which should bring various benefits to the society. Nonetheless, programming is usually considered to be a complex skill and many higher education institutions (HEIs) are faced with the problem of students struggling to complete their courses on introductory programming.

Any novel teaching method or technique that aims to improve learning of programming would be more valued if it could be successfully applied in different courses and across multiple countries. However, each course is unique in terms of its students, teachers, and the overall educational context in which it is organized. In addition to national regulations and HEI policies, teacher's decisions and sometimes even those of students may affect the structure, requirements, and outcomes of a course. Still, course comparison, which may not be an easy task even for two courses on the same subject, could provide useful information when devising improvements in teaching. If an improvement is to be generally applicable, it would need to be evaluated in different environments that have already been studied and sufficiently understood

In the present study, we examine and compare two groups of courses in higher education that are devoted to introductory programming, a group of courses from HEIs

in Portugal and a group of courses from an HEI in Serbia. This study represents an initial phase of a bilateral research project whose aim is to propose improvements to information technology (IT) teaching practice in Portugal and Serbia. In this initial phase, we are concerned with collecting anonymised data about selected courses on introductory programming, processing the data to obtain comparable data sets, exploring and contrasting the data sets, and discovering more about potential challenges in course execution in particular countries. Once the critical areas of improvement are identified, we may design concrete interventions and perform evaluation, which is the subject of subsequent phases of our project.

II. RELATED WORK

Portugal and Serbia are European countries of comparable size and population. Unlike Portugal, Serbia is not part of the European Union (EU), but it has entered the negotiation process to become a member. Nevertheless, the use of ICT in enterprises in Serbia is comparable to the corresponding level in the EU and, for some indicators, above the EU average [1].

Both countries are part of the Bologna process and are full members of the European Higher Education Area (EHEA) [2]. As a result, in these countries, the higher education systems are organized around the three-cycle format (bachelor's, master's, and doctoral degrees). Moreover, science, technology, and engineering appear to be prominent areas of higher education in the two countries. In Portugal, the annual number of students who completed their studies in areas related to science and engineering has exhibited a relatively positive trend over the past few years [3]. In Serbia, the official policies have been supportive of higher education in engineering and natural sciences [4].

Despite the fact that programming is currently a lucrative skill that can be acquired at a great number of HEIs, students tend to associate the process of learning to program with various difficulties. Based on responses from five different European countries, it may be inferred that students who learn to program overestimate their understanding and that more attention should be given to

certain programming topics [5]. Moreover, in a different study, it was observed that two introductory programming courses in Portugal and the United Kingdom had some common issues during course execution, including poor lecture attendance and problems in laboratory classes [6].

Difficulties in introductory programming courses have been categorized as being related to the nature of programming, teaching methods, study methods, student abilities and attitudes, or psychological challenges faced by students [7]. For instance, in a survey of students in Portugal, lack of effort or personal persistence and lack of motivation were some of the common reported reasons for learning difficulties [8]. In other studies involving students from Portugal, it was noted that final marks in an introductory programming course were positively correlated with final marks in some mathematics courses [9] and that students who failed an introductory programming course also had weak understanding of basic mathematical concepts and struggled with problem solving [10].

According to the reports in numerous studies, some of which have been mentioned in this section, it is evident that introductory programming is demanding and that students and teachers may face a varied set of issues in programming education. Before intervening in a course, it is appropriate to first compile and analyse data about concrete courses and then to identify issues in the courses.

III. DATA AND METHODS

For the purpose of our study, we collected and used anonymised data about a selection of introductory programming courses in higher education in Portugal and Serbia. Student identifiers were not part of the collected data. We analysed a period spanning academic years from 2013/2014 to 2016/2017. We have collected data for two institutions in Portugal, coded as PI-A and PI-B, and one institution in Serbia, coded as SI-A. In the context of our study, the term institution was used to denote an HEI at the level of a university or an institute.

We analysed the following courses:

- courses PC-A1 through PC-A7 – seven courses from the institution PI-A that belong to four different study programmes in total;
- courses PC-B1 through PC-B5 – five courses from the institution PI-B that belong to three different study programmes in total; and
- courses SC-A1 through SC-A3 – three courses from the institution SI-A that belong to three different study programmes in total.

For each execution of an analysed course, we recorded academic year, course code, institution code, programming language taught in the course, number of registered students, values of three indicators of student performance, and teaching and assessment methods that were predominantly used in the course.

The three indicators of student performance were:

- Passed/Assessed – the pass rate when considering only assessed students, i.e., a ratio of the number

of students who passed the course and the number of students who were assessed in the course;

- Passed/Registered – the pass rate when considering all students, i.e., a ratio of the number of students who passed the course and the number of students who registered for the course; and
- Assessed/Registered – the assessment rate, i.e., a ratio of the number of students who were assessed in the course and the number of students who registered for the course.

The same three indicators were used in a different study as metrics to assess the student performance in Portugal across consecutive academic years [11]. The definition of the state of being assessed actually differed between the two countries. For the analysed courses in Serbia, i.e., courses SC-A1 through SC-A3, a student was considered assessed only if the student had earned more than zero points in at least two different assessments. This definition was adopted because courses SC-A1 through SC-A3 had multiple ordinary pre-exam assessments throughout the term, unlike the analysed courses in Portugal. Due to limitations of the available data about the selected courses in Serbia, the values of the three performance indicators had to be calculated differently for the courses SC-A1 through SC-A3, i.e., only the students who were enrolled in the course for the first time were considered and the values of the indicators relying on the number of assessed students had to be based on certain estimates.

We considered the following teaching methods:

- 1 – doing paper-based exercises, i.e., students do not solve exercises by using computers, but by using pencil and paper;
- 2 – doing computer-based exercises, i.e., students use computers to solve exercises, either by using an editor and a command line compiler or by using an integrated development environment (IDE);
- 3 – project-based learning (PBL), i.e., projects are developed during the programming classes and programming concepts are introduced when needed in the project;
- 4 – using learning management systems (LMS), i.e., a software platform is used to manage classes and work outside the classroom;
- 5 – using syntax-oriented editors, i.e., syntax-oriented editors are used when programming in the classroom; and
- 6 – using debuggers, i.e., debuggers are used when programming in the classroom.

We considered the following assessment methods:

- 1 – project-based assessment, i.e., students produce some code to solve a problem and then send or present the code to the teacher;
- 2 – individual work, i.e., code is produced by just one student and there is not any group work;

- 3 – midterm assessment with paper-based exercises, i.e., assessment at the middle of the academic term when students use pencil and paper to solve exercises;
- 4 – midterm assessment with computer-based exercises, i.e., assessment at the middle of the academic term when students write programs on computers in the classroom;
- 5 – exam assessment, i.e., students do an exam at the end of the academic term, either by using computers or by pencil and paper;
- 6 – automated assessment, i.e., the teacher uses a special tool to automatically assess student code;
- 7 – peer assessment, i.e., student’s code is assessed by other students; and
- 8 – multiple assessments, i.e., students have multiple assessments throughout the academic term.

Multiple teaching and multiple assessment methods may be applied during the same course execution.

IV. RESULTS AND DISCUSSION

In Table 1, there are some basic data about each selected course, most notably the programming language taught and employed teaching and assessment methods. The dominant programming language is C, followed by Haskell and MATLAB. The dominance of the C language in programming courses coincides with the findings of other researchers. In a separate study of programming courses in the European higher education, it was concluded that C is the language most often taught, with strong advantage over the other languages especially in the first term [12]. In general, the preference for C in programming education is typical of Serbia, but, in Portugal, Java and Python are actually used more often than C to teach programming [12].

In both countries, teaching methods involved computer-based exercises and, to a lesser degree, usage of syntax-oriented editors. Another commonality is that paper-based exercises were not used in teaching, i.e., this traditional method has been abandoned in practice. However, for the analysed courses, it may be also noticed that some new trends, such as learning management systems, peer assessment, and even fully automated assessment of student code, have not been adopted yet.

In general, student project work has become more common in higher education, especially in engineering and science studies. In the context of the considered courses, project-based learning and project-based assessment were applied only at one institution in Portugal, i.e., solely within the courses at the institution PI-A. We may only hypothesise that the absence of project-based assessment at the other two institutions is related to the insufficient number of academic staff members needed for such a demanding task.

TABLE I. OVERVIEW OF COURSES

Institution code	Course code	Language taught	Teaching methods	Assessment methods
PI-A	PC-A1	Haskell	2, 3	1, 3, 4, 5
PI-A	PC-A2	Haskell	2, 3	1, 3, 4, 5
PI-A	PC-A3	Haskell	2, 3	1, 3, 4, 5
PI-A	PC-A4	C	2, 3	1, 3, 4, 5
PI-A	PC-A5	C	2, 3	1, 3, 4, 5
PI-A	PC-A6	C	2, 3	1, 3, 4, 5
PI-A	PC-A7	C	2, 3	1, 2, 5
PI-B	PC-B1	C	2, 5	3, 4, 5
PI-B	PC-B2	C	2, 5	3, 4, 5
PI-B	PC-B3	C	2, 5, 6	3, 4, 5
PI-B	PC-B4	C	2, 5	3, 5
PI-B	PC-B5	MATLAB	2, 5	3, 4, 5
SI-A	SC-A1	C	2, 5	2, 5, 8
SI-A	SC-A2	C	2, 5	2, 5, 8
SI-A	SC-A3	C	2, 5	2, 5, 8

There was also some variation between the choices of other assessment methods. There was an exam assessment in both countries, but the execution of assessment throughout the term differed. In Portugal, there was almost always some form of midterm assessment, while, in Serbia, there were only multiple assessments organized throughout the term.

In Fig. 1, the values of the Passed/Registered ratio are given across the examined academic years for all the considered courses. The top and middle chart in Fig. 1 show that the two institutions in Portugal have similar pass rates, which could be classified as low to moderate. On the other hand, the Passed/Registered ratio is generally better in courses in Serbia and the pass rates there could be regarded as high. As the compiled data do not include assessment scores and final grades of students, we could not speculate if the higher pass rates were the result of potentially better student knowledge of programming in Serbia. In a global survey of pass and failure rates in introductory programming courses, the reported average pass rate was 72% when all the courses were weighted equally [13]. With respect to this value, the pass rates could be considered above average for the institution in Serbia and below average for the institutions in Portugal.

At least some portion of the difference in pass rates between Portugal and Serbia could be attributed to the fact that all the performance ratios for the courses in Serbia were calculated only for the students who were registered for the course for the first time in a particular year. Nonetheless, based on the knowledge of past course executions in Serbia, it may be supposed that, even if all the registered students were considered, the pass rates in Serbia would on average still be higher than in Portugal. In this other scenario, the pass rates in Serbia would probably move towards the moderate to high category.

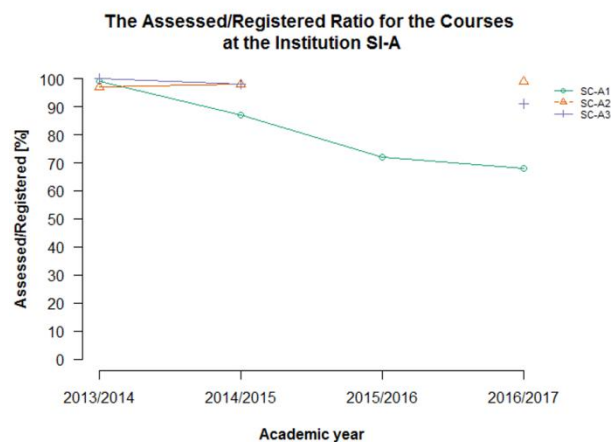
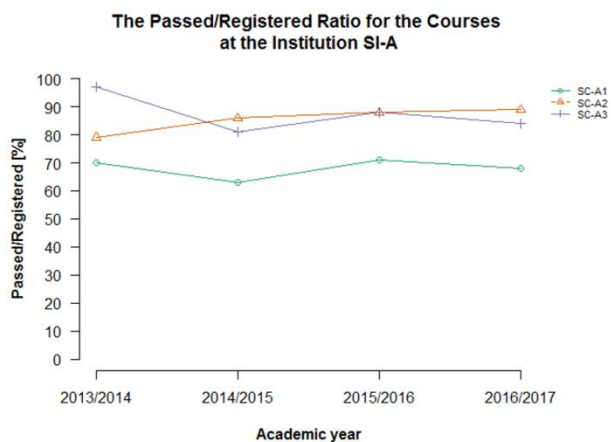
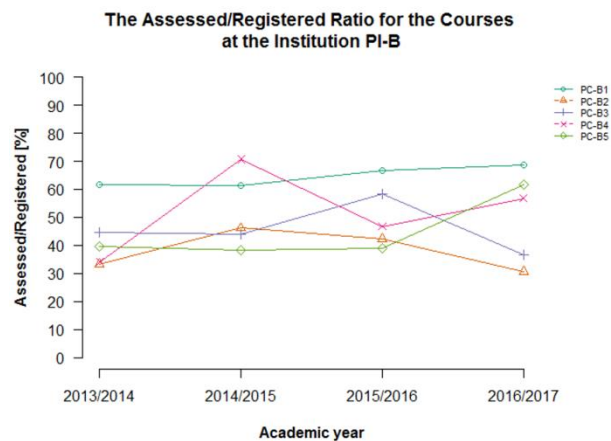
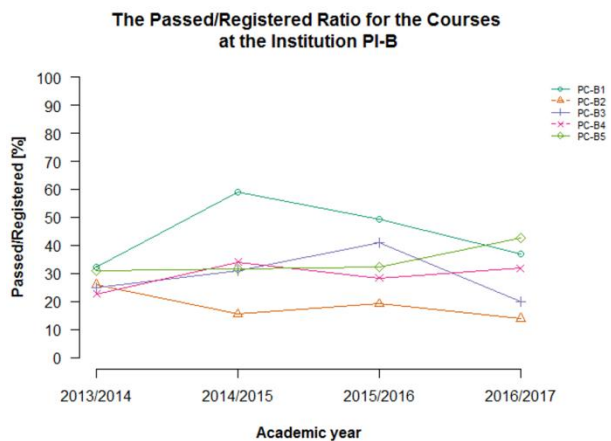
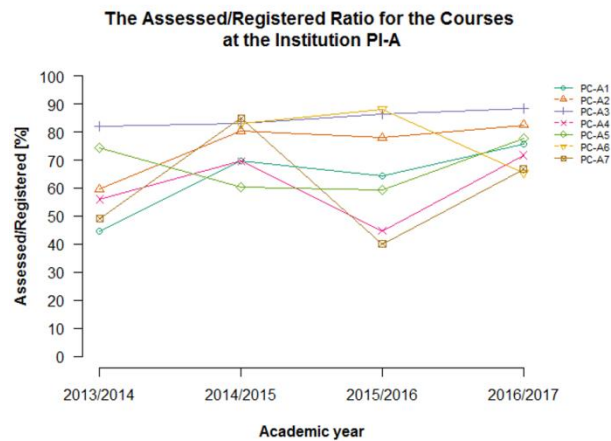
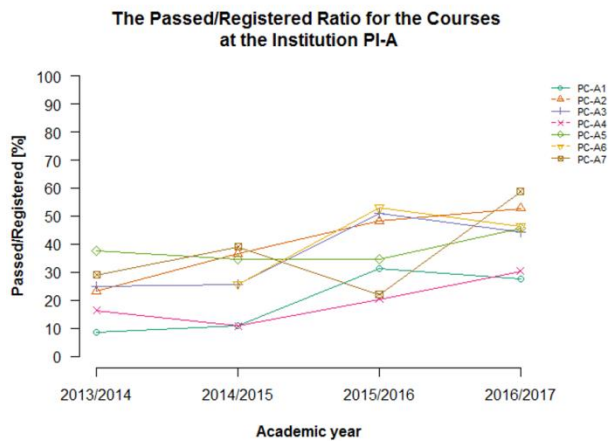


Figure 1. The Passed/Registered indicator values for different institutions: PI-A (top), PI-B (middle), and SI-A (bottom)

Figure 2. The Assessed/Registered indicator values for different institutions: PI-A (top), PI-B (middle), and SI-A (bottom)

In Fig. 2, the values of the Assessed/Registered ratio are shown. As expected, the values of the Assessed/Registered ratio are somewhat higher than the corresponding values of the Passed/Registered ratio. The differences in the Assessed/Registered ratio between the institutions in Portugal and Serbia are less pronounced than for the Passed/Registered ratio. However, these differences are still noticeable and the institution in Serbia exhibits higher assessment rates. The previous commentary regarding the influence of sample construction on the Passed/Registered values may also

apply to the case of interpreting the Assessed/Registered values.

The greatest difference between the Passed/Registered and Assessed/Registered values may be observed for the institution PI-A, which is located in Portugal. Such a difference may be more easily observed in Fig. 3, where the values of the Passed/Assessed ratio are presented. Based on these values, it may be presumed that the Passed/Assessed ratio is noticeably higher in Serbia as opposed to Portugal.

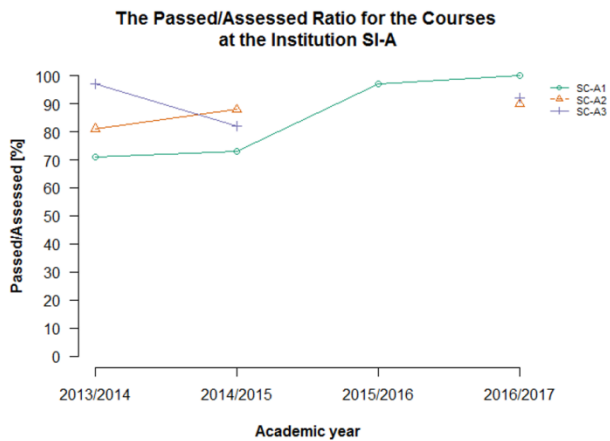
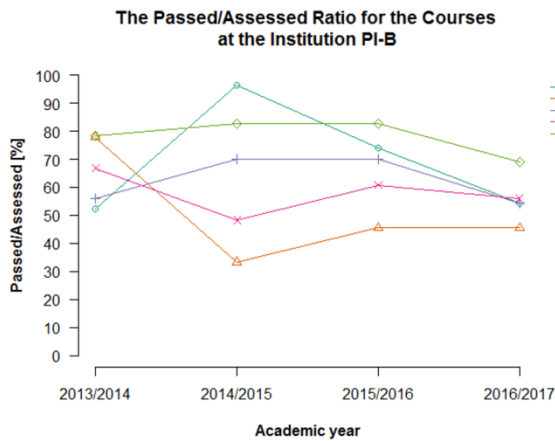
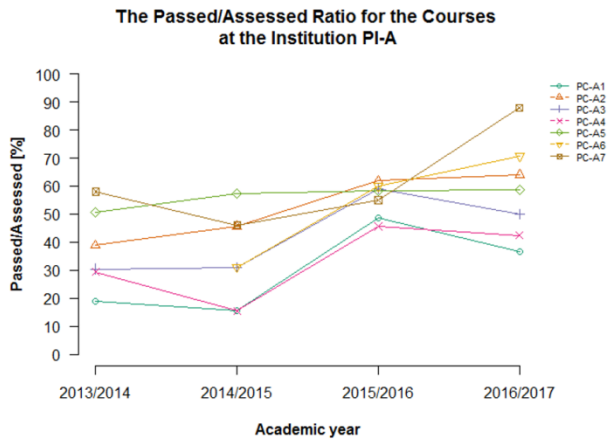


Figure 3. The Passed/Assessed indicator values for different institutions: PI-A (top), PI-B (middle), and SI-A (bottom)

V. CONCLUSION

We collected and analysed anonymised data about selected introductory programming courses at HEIs in Portugal and Serbia. The main findings of the comparative analysis of the courses in the two countries include:

- the C language is the main choice for introductory programming courses in both Portugal and Serbia;
- teaching methods differ between the countries – computer-based exercises are used in both countries, but PBL is applied only in Portugal;

- assessment methods differ between the countries – exams were applied in both countries, but midterm assessment is usually preferred in Portugal and multiple assessments throughout the term are preferred in Serbia;
- assessment rates are higher in Serbia; and
- pass rates are higher in Serbia, even when only the assessed students are considered.

The most prominent distinction between the courses in the two countries is the difference in both the pass rates and the assessment rates. Potential causes might include differences in teaching methods and course requirements, as well as disparities in motivation and attitudes of students. Another factor might be the lack of previous education in computing and the absence of prerequisite skills. However, the introduction of computer science into basic education [14] could contribute to better preparedness of future students, which may reduce the present problems. Since we are interested in improving current IT teaching practice, we are planning to look more closely into the identified issues.

ACKNOWLEDGMENT

The research was conducted as part of the project *Data mining based evaluation of IT teaching practice in Portugal and Serbia* within the Programme of Scientific and Technological Cooperation between the Government of the Portuguese Republic and the Government of the Republic of Serbia, and also supported by the Ministry of Education, Science, and Technological Development of the Republic of Serbia under Grant III-44010.

REFERENCES

- [1] A. Zečević and J. Radović-Stojanović, "The use of information and communication technologies in enterprises in Serbia," *Ekonomika preduzeća*, vol. 65, no. 5-6, pp. 393-402, Sep.-Oct. 2017.
- [2] European Commission/EACEA/Eurydice, "The European Higher Education Area in 2018: Bologna Process Implementation Report," Publications Office of the European Union, Luxembourg, 2018. [Online]. Available: https://eacea.ec.europa.eu/national-policies/eurydice/sites/eurydice/files/bologna_internet_0.pdf, Accessed on: Sep. 15, 2018.
- [3] PORDATA, "Diplomados no ensino superior: total e por área de educação e formação," (in Portuguese), Jul. 27, 2018. [Online]. Available: <https://www.pordata.pt/Portugal/Diplomados+no+ensino+superior+total+e+por+C3%A1rea+de+educa%C3%A7%C3%A3o+e+forma%C3%A7%C3%A3o-222>, Accessed on: Sep. 15, 2018.
- [4] V. Ivančević, V. Ivković, and I. Luković, "Open data in an analysis of higher education in engineering and technology in Serbia," in 45th SEFI Annual Conference, Azores, Portugal, 2017, pp. 1260-1267.
- [5] E. Lahtinen, K. Ala-Mutka, and H.-M. Järvinen, "A study of the difficulties of novice programmers," in 10th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education, Caparica, Portugal, 2005, pp. 14-18.
- [6] I. Huet, O. R. Pacheco, J. Tabares, and G. Weir, "New challenges in teaching introductory programming courses: A case study," in 34th Annual Frontiers in Education Conference, Savannah, GA, USA, 2004, pp. T2H/5-T2H/9.

- [7] A. Gomes and A. J. Mendes, "Learning to program - Difficulties and solutions," in International Conference on Engineering Education, Coimbra, Portugal, 2007.
- [8] A. J. Gomes, A. N. Santos, and A. J. Mendes, "A study on students' behaviours and attitudes towards learning to program," in Annual Conference on Innovation and Technology in Computer Science Education, Haifa, Israel, 2012, pp. 132-137.
- [9] A. J. Gomes and A. J. Mendes, "A study on student performance in first year CS courses," in 15th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education, Ankara, Turkey, 2010, pp. 113-117.
- [10] A. Gomes, L. Carmo, E. Bigotte, and A. Mendes, "Mathematics and programming problem solving," in 3rd E-learning Conference – Computer Science Education, Coimbra, Portugal, 2006.
- [11] A. J. Mendes, L. Paquete, A. Cardoso, and A. Gomes, "Increasing student commitment in introductory programming learning," in 42nd Annual Frontiers in Education Conference, Seattle, WA, USA, 2012, pp. 82-87.
- [12] V. Aleksić and M. Ivanović, "Introductory programming subject in European higher education," *Informatics in Education*, vol. 15, no. 2, pp. 163-182, 2016.
- [13] J. Bennedsen and M. E. Caspersen, "Failure rates in introductory programming," *ACM SIGCSE Bulletin*, vol. 39, no. 2, pp. 32-36, Jun. 2007.
- [14] D. Passey, "Computer science (CS) in the compulsory education curriculum: Implications for future research," *Education and Information Technologies*, vol. 22, no. 2, pp. 421-443, Mar. 2017.