Automatic Feedback in the Teaching of Programming in Undergraduate Courses: a Literature Mapping

Wanderson Conceição
Dept. of Computer Science
University of Brasilia
Brasilia, Brazil
0000-0002-3663-7017

Maristela Holanda
Dept. of Computer Science
University of Brasilia
Brasilia, Brazil
0000-0002-0883-2579

Fernanda Macedo
Dept. of Computer Science
University of Brasilia
Brasilia, Brazil
0000-0003-2055-0424

Edison Ishikawa
Dept. of Computer Science
University of Brasilia
Brasilia, Brazil
0000-0002-0214-9234

Vanessa Tavares Nunes
Dept. of Computer Science
Fluminense Federal University
Rio de Janeiro, Brazil
0000-0002-5043-0217

Dilma Da Silva
Dept. of Computer Science and Engineering
Texas A&M University
College Station, US
0000-0001-6538-2888

Abstract— Teaching programming in the early years of undergraduate courses has been a challenge for students, institutions, and professors. In view of this, Learning Management Systems (LMSs) and other teaching platforms have emerged to address some of the difficulties in this process. In this context, the present work intends to answer the following research question (RQ): What does the literature tell us about the use of automatic feedback in teaching programming in undergraduate courses? To answer this question, a literature mapping was conducted based on 119 articles published between 2017 and 2021. The mapping showed that the research area is expanding and has related studies from all over the world. The papers have different origins, and 37 countries are represented in this survey. The main programming languages used are Java, Python, C and C++. Another finding was that it is common practice to develop specific platforms for automatic feedback in programming courses. This paper presents the findings and results obtained.

Keywords — automatic feedback, undergraduate, programming, computer science education

I. Introduction

Given the prevalent role of technology in society, teaching programming in the initial years of undergraduate courses has become critical, especially in computer science courses [2]. However, the process of learning the first programming language is seen, by most students, as a challenge, especially for beginners [1]. Such difficulty in assimilating the concepts and structures inherent in computer programming can be interpreted as one of the reasons why failure rates in computer science courses are so high [2].

Therefore, Learning Management Systems (LMS) and other teaching platforms emerge to assist in the teaching-learning process, but most of these environments do not have the necessary dynamism, which is intrinsic to the teaching-learning process and, especially, to the teaching of the first programming language [3].

Tools and virtual environments created to help instructors and students have been present in the educational environment since the 1960s [4]. These virtual environments have a methodology based on the visualization of feedback or tips through textual information, which aims to guide the student to a certain stage of the learning process [1][5].

In this context, this paper presents a literature mapping that intends to answer, in general, the following research question: What does the literature tell us about the use of automatic feedback in the teaching of programming in undergraduate courses? To facilitate understanding and get a better insight into the topic and its related issues, the central inquiry has been divided into seven sub-questions, which together will answer the suggested central problematic, using systematic mapping [11] of recent literature to obtain the answers.

Searches were made using a unified search string in the Scopus and Web of Science academic databases, followed by the application of the inclusion and exclusion criteria, resulting in a set of 119 papers selected for analysis to answer the proposed questions.

This paper is divided into five more sections, which are: Section II, where the methodology and research questions are described; Section III, which details the paper selection process; Section IV, where the research questions are answered; Section V, where the results obtained are discussed, and finally, the conclusions and future work are presented in Section VI.

II. Research Questions

In the present paper, a literature mapping was carried out. The procedures performed are presented in order to clarify the steps to the results and conclusions. The mapping process was based on [11], which is defined in four steps: *i)* definition of the research questions; *ii*) selection of the most relevant works within the search; *iii*) analysis of the selected works and finally *iv)* answers to the research questions. The core of this paper is the main research question (RQ): What does the literature tell us about the use of automatic feedback in the teaching of programming in undergraduate courses? To answer this question, seven related questions were created:

- RQ1) What is the number of papers published per year related to this topic?

- RQ2) Which countries have produced most of these papers?
- RQ3) Which information sources (conferences or journals) published the most papers on this topic?
- RQ4) What are the main LMSs used?
- RQ5) What are the main programming languages used?
- RQ6) Do the papers present any kind of perception evaluation of those who make use of the automatic feedback?
- RQ7) What type of response is presented in the feedback (Constructive, Informative, Corrective, Motivational, etc.)?

The steps of the literature mapping will be explained under the following headings.

III. Selecting the Relevant Papers

The stage of selecting the relevant papers is fundamental in any literature mapping process. It involves defining a search string, choosing the academic databases in which to search for the papers, and specifying the inclusion/exclusion criteria to decide the significance of each paper in the study.

To set the search string, the first criterion was to identify keywords directly related to the theme. Then, synonyms were added to the search in order to optimize the query. The first version of the search string was:

(“feedback” or “response” or “answer”) AND (“automatic” or “automated”) AND “computer” AND “programming”.

This first draft of the search string was limited by the areas (“computing” and “engineering”) and achieved results in its first version: 1,960 results in Scopus in November 2021. We constrained the area because “feedback” is linked to other fields of study, such as the social sciences, business administration, and psychology.

Several queries in the selected bases were made from this initial search string. During the refinement of the search and the string assembly, the tool VOSviewer [110] was used to analyze the words used in the query formation and make associations, through heat maps, with other words that were potentially related to the theme. VOSviewer was of utmost importance to increase the accuracy of the search string, offering several keywords that were not considered during the construction of the expression. The result of the final search string was as follows:

("feedback" OR "response" OR "answer") AND ("programming" OR "programming languages" OR "code" OR "coding" OR “CS1" OR "ICS") AND ("students" OR "undergraduate" OR "novice" OR "freshman”) AND ("teaching" OR "learning" OR "e-learning")

The academic databases selected in this mapping are Scopus and Web of Science because they index the most important conferences related to Computer Education (such as IEEE FIE and ACM SIGCSE) and the ACM and IEEE journals.

As inclusion and exclusion criteria, some metrics were defined to be applied during the base search. These were the inclusion criteria:

- IC1: The papers should only refer to the areas of computing and engineering;
- IC2: Papers must have a publication date between the years 2017 and 2021;
- IC3: Papers must be written in English;
- IC4: Papers should be of the type "Journal Article" or "Conference Paper".

Figure 1 provides a representation of the research cycles. The use of the main string as well as the application of the criteria occurred between February and March 2022. Initially, it returned 1,587 results in the Scopus database and 564 results in the Web of Science, totalizing 2,151 selected articles and 393 duplicates (Table I).

TABLE I. Quantity Of Papers Per Database

Source	***Number of Papers***
Scopus	1587
Web of Science	564
Duplicates	393
Total	1758

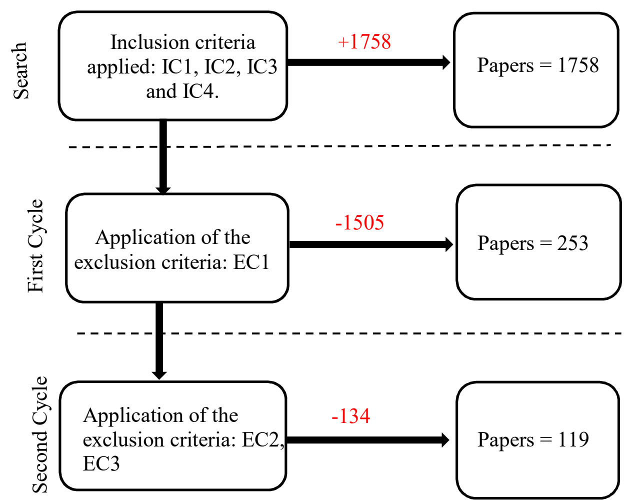

Fig. 1. Synopsis of Inclusions and Exclusions and their respective criteria applied at each stage of the research.

A. First Cycle

The total of papers filtered in the search (1758) was used in the first analysis cycle. After removing duplicate fields, a spreadsheet with the following metadata was created: database source, title, abstract, year of publication, authors, author keywords, document type, and publication venue.

In an attempt to exclude those papers that were not in accordance with the research topic, the exclusion criterion was applied:

- EC1: Remove all papers that are not related to the use of automatic feedback in programming courses.

Most of the papers filtered previously were removed, as they were mainly about the use of feedback in other contexts, for example, in issues related to social sciences, behavior, coexistence and other areas not related to the proposed theme, thus ending the first cycle with 253 papers selected.

B. Second cycle

The second cycle of selecting relevant papers consisted of applying exclusion criteria EC1 to further filter papers related to the topic of using automatic feedback to teach programming. The following criteria were used along with EC1 for filtering papers:

- EC2: papers with less than 3 pages;
- EC3: papers that deal with automatic feedback but in other areas of computing.

As an initial step, the application of EC2 resulted in the exclusion of 7 papers from the 253 results remaining after applying EC1. These 7 papers were short experience reports or short papers on the use or development of feedback systems.

Next, following the application of EC3, another 134 papers were removed. It is important to note that despite the fact that the papers had been filtered by area (computing and engineering), some of them, although within the delimited areas, discussed cognitive or behavioral issues, or the feedback was not applied to the teaching of programming.

A significant obstacle encountered during the search and selection of works was finding papers unrelated to the social sciences and focused only on the area of computing. Because it is a broad theme and reported in several contexts, the process of application of EC3 criteria to the results obtained was done manually.

After this process, 119 papers that dealt with the proposed theme were selected, thus finalizing the search cycle.

IV. Results and Analysis

This section answers the seven research questions based on the selected 119 articles.

A. RQ1: What is the number of papers published per year related to this topic?

As seen in Figure 2, the number of publications per year had a significant growth between the years 2017 and 2019. Notably, in the subsequent years, 2020 and 2021, the number of publications fell, returning to the 2018. This period coincided with the pandemic, so the significant disruptions in educational and research activities may have impacted the number of publications. Even with the drop in the number of publications, it is possible to see that the rates are still constant, as shown in the trend line.

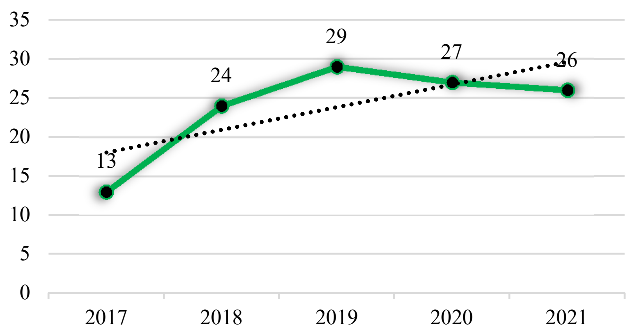

Fig. 2. Number of papers published in the last 5 years.

B. RQ2: Which countries have produced most of these papers?

During the review process, 37 countries and 5 continents were identified: America, Oceania, Europe, Asia and Africa. Figure 3 shows the countries that are present in the selection results.

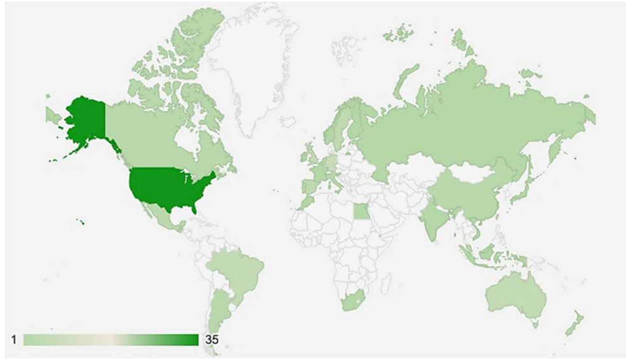

Fig. 3. Distribution across countries of publications related to automatic feedback in programming courses.

The 16 countries that had the largest production of papers on the subject are presented in Table II. The countries that produced the most papers are the United States and Germany. The following countries also appear, although with smaller numbers: Taiwan, United Kingdom, Netherlands, Australia, Spain and Sweden.

Table II. Countries with the most publications.

Countries	***Nº***	***Papers***
United States	34	[1][4][8][3][6][12][21][23][26][30][33][47][49][50][54][68][71][74][75][76][77][78][81][83][89][90][92][94][98][102][111][112][115][121][122]
Germany	7	[5] [18] [34][56] [63] [64][104]
Taiwan	6	[22] [55] [57] [103] [116] [117]
Australia	5	[27][40][79][95][107]
United Kingdom		[19][29][53][84]
Netherlands	4	[35][58][61][97]
Spain	4	[8][14][38][45]
Sweden	4	[41][42][43][44]
Belgium	3	[17][28][70]
Brazil	3	[7][101][119]
Mexico	3	[60][65][106]
Portugal	3	[36][73][123]
Singapore	3	[2][32][124]
Argentina	2	[13][125]
Canada	2	[99][120]
China	2	[31][114]
Estonia	2	[9][85]
Italy	2	[15][39]
Japan	2	[72][105]
Malaysia	2	[16][126]
New Zealand	2	[25][113]

Figure 3 and Table II show the United States' preeminence: American institutions produced 35 of the 119 articles analyzed, representing 29.41% of the relevant publications related to the theme.

As additional information to the data in Table II, the following countries had one publication on the topic in the last five years: Djibouti [52], Egypt [86], Finland [46], France [51], Guatemala [20], Hong Kong [69], India [82], Indonesia

[93], Ireland [118], Israel [118], Morocco [10], Norway [80], Russian Federation [100], Slovenia [59], South Africa [108] and Vietnam [67].

C. *RQ3) Which information sources (conferences or journals) published the most papers on this topic?*

Table III provides information on the conferences and journals. Most of the publication venues are classified as conferences, with 11 articles from journals and 108 papers from conference proceedings. Thus, the conference papers correspond to 94.2% of the papers, the journals correspond to 7.5% of the selected publications.

TABLE III. CONFERENCES AND JOURNALS.

Venue	***Nº***	***Papers***	***Type***
IEEE Frontiers In Education, (FIE)	12	[7][12][28][44] [49][54][85] [91][92][98] [120]	C
Special Interest Group Computer Science Education (SIGCSE) Technical Symposium	7	[13[23] [48] [64] [68] [90] [115]	C
Conference on Integrating Technology into Computer Science Education (ITiCSE)	5	[40][41] [77][96][102]	C
International Computing Education Research (ICER)	5	[47] [74][76] [81][122]	C
International Conference on Software Engineering (ICSE)	3	[2][95][124]	C
Annual Conference and Exposition, Conference Proceedings (ASEE)	3	[6] [37] [65]	C
Computer Applications in Engineering Education (CAEE)	3	[45][67][119]	J
Global Engineering Education Conference (EDUCON)	3	[56] [63][80]	C
IEEE Access (IEEE A)	3	[29] [38] [46]	J
Conference on Learning @ Scale	3	[21] [69] [104]	C
Educational Data Mining in Computer Science Education (CSEDM)		[3][75][78]	C
International Conference on Intelligent Tutoring Systems (ITS)	3	[97][101][105]	C
European Conference on Technology Enhanced Learning (ECTEL)	3	[18][35][118]	C
International Conference on Artificial Intelligence in Education (ICAIE)	2	[1][79]	C
International Conference on Teaching, Assessment, and Learning for Engineering (TALE)	2	[27][53]	C
International Conference on Educational Data Mining (EDM-WS)	2	[89][111]	C
International Conference on Educational and Information Technology (CEIT)	2	[4] [116]	C
International Symposium on Computers in Education (SIIE)	2	[73][123]	C

To complement the information in Table III, it is important to mention that: 31 conferences with only one paper were identified, among which we can mention ACM Special Interest Group on Computer-Human Interaction, SIGCHI[39], IEEE Integrated STEM Education Conference, ISEC [20]. Regarding journals, there were 5 articles with only one publication, in this group we can mention ACM Transactions on Computing Education, TOCE[61], IEEE Transactions on Learning Technologies, IEEE TLT[14] and Journal of Internet Technology, JIT[55] as relevant.

D. *RQ4) What are the main LMSs used?*

From this study, it may be noted that the LMSs used are not standardized tools or structures; institutions sought to develop their own tools or adopted systems already available in the market.

Based on the selected papers, it seems to be a common practice to name the platforms developed for automatic feedback in the teaching of programming. Even knowing this apparent convention, of 119 papers analyzed, about 29 papers did not make it clear which types of environments they used in their research. Another 15 papers had not yet started developing their respective platforms, so they were not considered as such.

As for the research that adequately identified systems, there was no specific system that was used in general by the authors. The explanation for this may be linked to the different problems that students and institutions seek to solve with the use of feedback in the teaching-learning process. In view of this, 75 of the 119 papers present some kind of tool, meaning a total of 63% of the publications have specific systems that use, in some way, automatic feedback (Figure 4).

According to [50], the use of automatic feedback is adaptive because, besides serving several areas, such an interaction tool can serve several types of audiences and needs. Such need for adaptability and customizability may explain the large number of platforms developed. Some widely deployed platforms such as GitHub [54] and Moodle [119] are used, but they are not common in our mapping.

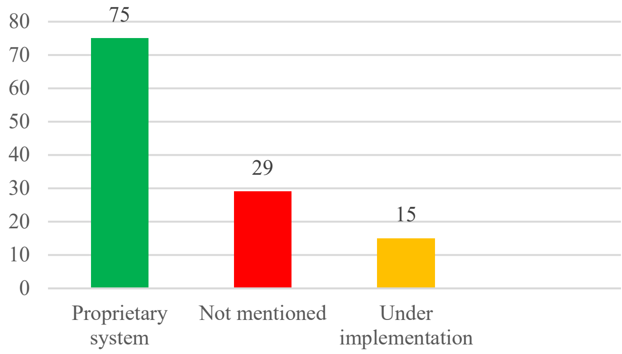

Fig. 4. Quantity of articles using LMS environments.

E. *RQ5) What are the main programming languages used?*

Among the languages found, some stood out, appearing in distinct feedback systems. This may indicate that these languages are frequently used in introductory computing courses (CS1). Because they are more used, it is natural that there are more tools geared towards them, made in a customized and optimized way. Table IV shows the main programming languages used in automatic feedback for teaching programming.

After checking the results obtained regarding the main programming languages used, it is important to point out that some automatic feedback tools go beyond interactions centered around coding. Many interactions, as in [20][62][77][80], use feature multiple choice questions as a reference for the feedback.

TABLE IV. THE MAIN PROGRAMMING LANGUAGES.

Languages	***Nº***	***Papers***
Java	18	[11][13][17][18][21][37][42][44][48][56][65][77][72][87][96][98][120]
Python	10	[12][14][17][19][57][62][88][99][101][119]
C	8	[41][71][73][78][81][93][100][114]
C++	7	[23][42][43][47][64][94][126]
NoCode	4	[24][33][39][55]
R	3	[83][91][107]
Haskell	1	[114]
Portugol	1	[121]

F. RQ6) Do the papers present any kind of perception evaluation of those who make use of the automatic feedback?

It was noted that a number of papers did not present any type of evaluation, be it of quality or perception, made by the student. In Figure 6, the percentage of papers that did not provide any kind of evaluation by the student is high, which is a possible drawback.

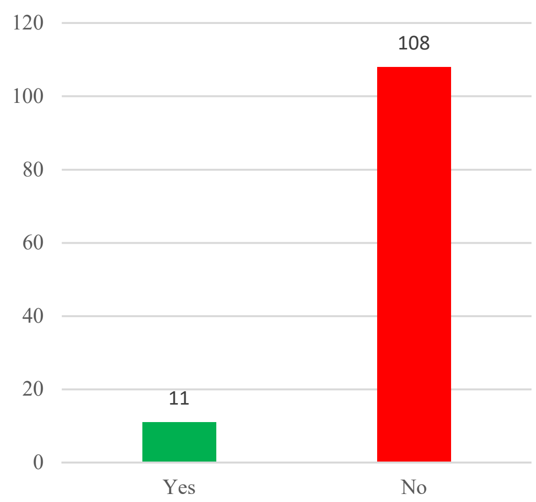

Fig.5. Perception evaluation of those who make use of the automatic feedback.

This student contribution seems to be of great importance to various aspects of the process of developing and improving feedback systems. This criticism can be a differential regarding the evolution of the feedback system being implemented.

G. RQ7) What type of response is presented in the feedback (Constructive, Informative, Corrective, Motivational, etc.)?

In the first half of the 20th century, the concept of feedback emerged in biology, electrical engineering, physics, and communication, with no consensus as to which field the term originated in [8] [10].

According to [8] [9], the main types of feedback that can be used in computing are motivational, positive, corrective, formative, summative and ipsative. Each type is described in Table V.

TABLE V. TYPES OF FEEDBACK.

Type	***Description***
Motivational	It is geared towards engagement and should encourage students to face obstacles and do their best [8][10].
Positive	Process of recognizing a student's performance or achievement [8][10].
Corrective	Particular datum to a student about a behavior, skill, or response that should be modified to meet desired goals/responses [8][9].
Formative	It seeks to modify the user's thinking and/or behavior in order to facilitate learning [8][9].
Summative	It evaluates a given test with the goal of assigning a grade to the student [8][9].
Ipsative	It compares the student or a response to an already defined pattern [8][9].

These types of feedback are classified based on the content of the response issued to the user. Table VI presents the feedback type in the relevant papers. The main types of feedback found in the papers are those that help the teacher to instill concepts, which can often be abstract, in the student's thinking. In this context, the formative and corrective types of feedback were the ones that stood out the most.

TABLE VI. TYPES OF FEEDBACK FOUND IN THE SEARCH.

Type	***Nº***	***Papers***
Formative	30	[7][12][19][24][23][33][37][38][41][42][43][44][45][47][58][66][71][72][73][74][77][78][81][87][98][101][107][110] [115][120]
Corrective	25	[1][11][13][14][17][22][27][29][31][39][41][43][46][48][57][70][85][89][92][98][99][103][116][119][126]
Motivational	1	[75]
Summative	3	[2][16][123]

The analysis of the types of feedback found showed a consistent behavior towards the undergraduate student, especially those who are learning their first programming language. Some papers did not classify the types of feedback used, some because they were not described, some because they were only about the software being developed and the development process, and some because they were other literature reviews like this one.

V. DISCUSSION

The use of automatic feedback in programming education is being researched all over the world and is growing steadily. With the evolution of computers and the access to the internet, and more recently with the great increase in distance learning courses during the last two years, because of the pandemic, it is expected that this theme will be studied even more.

The papers published in the area, despite covering almost all continents, are concentrated in the United States. In addition, FIE and SIGCSE were the conferences with the largest number of articles analyzed.

The publications considered here showed relevant contributions to using automatic feedback in teaching programming to undergraduate students. Most of the papers were about platforms developed by research groups being implemented in early undergraduate or programming courses. Java is the most mentioned programming language in the selected papers.

The analysis of the papers made it clear that no platform or method is 100% suitable, and it is up to the researcher and the institution to develop ideas for a platform/service that meets local and specific needs. In conclusion, there was no environment t not modified to meet the needs of the institution so far.

VI. Conclusion

This paper provides a literature mapping of the use of automatic feedback in undergraduate introductory programming courses. Our analysis showed that the use of feedback is being researched worldwide and that new tools are emerging to solve the problems related to teaching programming.

The results and answers obtained in this work make it relevant because they attempt to answer questions that have been little addressed over the years on the subject and contribute to the research by answering a reasonable number of questions.

In future work, this review will inform the development of an educational assistant tool in the University of Brasilia, Brazil, aiming to assist instructors and students in the programming courses offered by the institution.

References

[1] Ahmed, I., Lubold, N., & Walker, E. (2018, June). ROBIN: using a programmable robot to provide feedback and encouragement on programming tasks. In International Conference on Artificial Intelligence in Education (pp. 9-13). Springer, Cham.

[2] Ahmed, U. Z., Srivastava, N., Sindhgatta, R., & Karkare, A. (2020, June). Characterizing the pedagogical benefits of adaptive feedback for compilation errors by novice programmers. In Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: Software Engineering Education and Training (pp. 139-150).

[3] Akram, B., Azizolsoltani, H., Min, W., Wiebe, E. N., Navied, A., Mott, B. W., ... & Lester, J. C. (2020). A Data-Driven Approach to Automatically Assessing Concept-Level CS Competencies Based on Student Programs. In CSEDM@ EDM.

[4] Albinson, P., Cetinkaya, D., & Orman, T. (2020, February). Using Technology to Enhance Assessment and Feedback: A Framework for Evaluating Tools and Applications. In Proceedings of the 2020 9th International Conference on Educational and Information Technology (pp. 241-246).

[5] Albrecht, E., Gumz, F., & Grabowski, J. (2018, June). Experiences in introducing blended learning in an introductory programming course. In Proceedings of the 3rd European Conference of Software Engineering Education (pp. 93-101).

[6] Al-Haj, S., Seliya, N., & Kemner, C. L. (2019, June). Pedagogical Assessment of Secure Coding in Student Programs. In 2019 ASEE Annual Conference & Exposition.

[7] Andrade, R., & Brunet, J. (2018, October). Can students help themselves? An investigation of students' feedback on the quality of the source code. In *2018 IEEE Frontiers in Education Conference (FIE)* (pp. 1-8). IEEE.

[8] Anfurrutia, F. I., Álvarez, A., Larrañaga, M., & López-Gil, J. M. (2018). Integrating formative feedback in introductory programming modules. *IEEE Revista Iberoamericana de Tecnologias del Aprendizaje*, *13*(1), 3-10.

[9] Annamaa, A., Suviste, R., & Vene, V. (2017, November). Comparing different styles of automated feedback for programming exercises. In *Proceedings of the 17th Koli Calling International Conference on Computing Education Research* (pp. 183-184).

[10] Arifi, S. M., Abbou, R. B., & Zahi, A. (2020). Assisted learning of C programming through automated program repair and feed-back generation. *Indonesian Journal of Electrical Engineering and Computer Science*, *20*(1), 454-464.

[11] Keele, S. (2007). Guidelines for performing systematic literature reviews in software engineering (Vol. 5). Technical report, Ver. 2.3 EBSE Technical Report. EBSE.

[12] Beck, P. J., Mohammadi-Aragh, M. J., Archibald, C., Jones, B. A., & Barton, A. (2018, October). Real-time metacognition feedback for introductory programming using machine learning. In 2018 IEEE Frontiers in Education Conference (FIE) (pp. 1-5). IEEE.

[13] Benotti, L., Aloi, F., Bulgarelli, F., & Gomez, M. J. (2018, February). The effect of a web-based coding tool with automatic feedback on students' performance and perceptions. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education* (pp. 2-7).

[14] Benotti, L., Martnez, M. C., & Schapachnik, F. (2017). A tool for introducing computer science with automatic formative assessment. IEEE Transactions on Learning Technologies, 11(2), 179-192.

[15] Bertagnon, A., & Gavanelli, M. (2020, December). MAESTRO: a semi-autoMAted Evaluation SysTem for pROgramming assignments. In *2020 International Conference on Computational Science and Computational Intelligence (CSCI)* (pp. 953-958). IEEE.

[16] BIMBA, Andrew Thomas et al. The Effects of Adaptive Feedback on Student's Learning Gains.

[17] Bogdanova, D., & Snoeck, M. (2019, September). Use of personalized feedback reports in a blended conceptual modelling course. In *2019 ACM/IEEE 22nd International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)* (pp. 672-679). IEEE.

[18] Burke, B., Weßeler, P., & Vrugt, J. T. (2018, September). A Programming Language Independent Platform for Algorithm Learning. In *European Conference on Technology Enhanced Learning* (pp. 652-655). Springer, Cham.

[19] Buyrukoglu, S., Batmaz, F., & Lock, R. (2017, August). A new marking technique in semi-automated assessment. In *2017 12th International Conference on Computer Science and Education (ICCSE)* (pp. 545-550). IEEE.

[20] Calderón, D., Petersen, E., & Rodas, O. (2020, August). SALP: A Scalable Autograder System for Learning Programming-A Work in Progress. In *2020 IEEE Integrated STEM Education Conference (ISEC)* (pp. 1-4). IEEE.

[21] Cassidy, C., Goldman, M., & Miller, R. C. (2018, June). Glanceable code history: Visualizing student code for better instructor feedback. In *Proceedings of the Fifth Annual ACM Conference on Learning at Scale* (pp. 1-4).

[22] Cheng, L. C., Li, W., & Tseng, J. C. (2021). Effects of an automated programming assessment system on the learning performances of experienced and novice learners. *Interactive Learning Environments*, 1-17.

[23] Cordova, L., Carver, J., Gershmel, N., & Walia, G. (2021, March). A Comparison of Inquiry-Based Conceptual Feedback vs. Traditional Detailed Feedback Mechanisms in Software Testing Education: An Empirical Investigation. In Proceedings of the 52nd ACM Technical Symposium on Computer Science Education (pp. 87-93).

[24] Costa, E., Fechine, J., Silva, P., & Rocha, H. (2016). Modelos de Feedback para estudantes em Ambientes Virtuais de Aprendizagem. Jornada de Atualização em Informática na Educação, 5(1), 1-38.

[25] Crow, T., Kirk, D., Luxton-Reilly, A., & Tempero, E. (2020, October). Teacher perceptions of feedback in high school programming education: a thematic analysis. In *Proceedings of the 15th Workshop on Primary and Secondary Computing Education* (pp. 1-6).

[26] Cruz, G., Jones, J., Morrow, M., Gonzalez, A., & Gooch, B. (2017, July). An AI system for coaching novice programmers. In International Conference on Learning and Collaboration Technologies (pp. 12-21). Springer, Cham.

[27] Cunningham-Nelson, S., Mohammadi-Aragh, M. J., Goncher, A., & Boles, W. (2018, December). Panel Session—Providing Automated and Individually Tailored Assessment Feedback. In *2018 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)* (pp. 1233-1234). IEEE.

[28] de Moffarts, G., & Combéfis, S. (2020, October). Challengr, a Classroom Response System for Competency Based Assessment and Real-Time Feedback with Micro-Contests. In *2020 IEEE Frontiers in Education Conference (FIE)* (pp. 1-4). IEEE.

[29] Demaidi, M. N., Gaber, M. M., & Filer, N. (2018). OntoPeFeGe: Ontology-based personalized feedback generator. *IEEE Access*, *6*, 31644-31664.

[30] DeNero, J., Sridhara, S., Pérez-Quiñones, M., Nayak, A., & Leong, B. (2017, March). Beyond autograding: Advances in student feedback platforms. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education* (pp. 651-652).

[31] Du, Y., Su, X., & Jiang, Y. (2020, October). Developing an Intelligent Personal Tutor to Support EFL Learning. In *Proceedings of the 4th International Conference on Computer Science and Application Engineering* (pp. 1-5).

[32] Elmaleh, J., & Shankararaman, V. (2020). A visual analytics tool for personalized competency feedback. Association for Information Systems.

[33] Emmons, S. R., Light, R. P., & Börner, K. (2017). MOOC visual analytics: Empowering students, teachers, researchers, and platform developers of massively open online courses. *Journal of the Association for Information Science and Technology*, *68*(10), 2350-2363.

[34] Fangohr, H., O'Brien, N., Hovorka, O., Kluyver, T., Hale, N., Prabhakar, A., & Kashyap, A. (2020, June). Automatic Feedback Provision in Teaching Computational Science. In *International Conference on Computational Science* (pp. 608-621). Springer, Cham.

[35] Fehnker, A., Mader, A., & Rump, A. (2021, September). Atelier–Tutor Moderated Comments in Programming Education. In *European Conference on Technology Enhanced Learning* (pp. 379-383). Springer, Cham.

[36] Figueiredo, J., & García-Peñalvo, F. (2021, October). Teaching and Learning Strategies for Introductory Programming in University Courses. In *Ninth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'21)* (pp. 746-751).

[37] Frasson, C. (2021, October). Ontology Reasoning for Explanatory Feedback Generation to Teach How Algorithms Work. In Novelties in Intelligent Digital Systems: Proceedings of the 1st International Conference (NIDS 2021), Athens, Greece, September 30-October 1, 2021 (Vol. 338, p. 239). IOS Press.

[38] Galan, D., Heradio, R., Vargas, H., Abad, I., & Cerrada, J. A. (2019). Automated assessment of computer programming practices: the 8-years UNED experience. IEEE Access, 7, 130113-130119.

[39] Galassi, A., & Vittorini, P. (2021, July). Automated feedback to students in data science assignments: improved implementation and results. In *CHItaly 2021: 14th Biannual Conference of the Italian SIGCHI Chapter* (pp. 1-8).

[40] Garcia, R., Falkner, K., & Vivian, R. (2019, July). Instructional Framework for CS1 Question Activities. In Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education (pp. 189-195).

[41] Glassey, R. (2018, July). Managing assignment feedback via issue tracking. In Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education (pp. 382-382).

[42] Glassey, R. (2019, May). Developing feedback analytics: Discovering feedback patterns in an introductory course. In *Proceedings of the ACM Conference on Global Computing Education* (pp. 37-43).

[43] Glassey, R. (2019, May). Developing feedback analytics: Discovering feedback patterns in an introductory course. In *Proceedings of the ACM Conference on Global Computing Education* (pp. 37-43).

[44] Glassey, R., & Bälter, O. (2020, October). Put the students to work: Generating questions with constructive feedback. In *2020 IEEE Frontiers in Education Conference (FIE)* (pp. 1-8). IEEE.

[45] González, J. A., López, M., Cobo, E., & Cortés, J. (2018). Assessing Shiny apps through student feedback: Recommendations from a qualitative study. *Computer Applications in Engineering Education*, *26*(5), 1813-1824.

[46] Grönberg, N., Knutas, A., Hynninen, T., & Hujala, M. (2021). Palaute: An Online Text Mining Tool for Analyzing Written Student Course Feedback. *IEEE Access*, *9*, 134518-134529.

[47] Gusukuma, L., Bart, A. C., Kafura, D., & Ernst, J. (2018, August). Misconception-driven feedback: Results from an experimental study. In *Proceedings of the 2018 ACM Conference on International Computing Education Research* (pp. 160-168).

[48] Haldeman, G., Tjang, A., Babeş-Vroman, M., Bartos, S., Shah, J., Yucht, D., & Nguyen, T. D. (2018, February). Providing meaningful feedback for autograding of programming assignments. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education* (pp. 278-283).

[49] Hao, Q., & Tsikerdekis, M. (2019, October). How automated feedback is delivered matters: Formative feedback and knowledge transfer. In *2019 IEEE Frontiers in Education Conference (FIE)* (pp. 1-6). IEEE.

[50] Hao, Q., Smith IV, D. H., Ding, L., Ko, A., Ottaway, C., Wilson, J., ... & Greer, T. (2021). Towards understanding the effective design of automated formative feedback for programming assignments. *Computer Science Education*, 1-23.

[51] Hodeib, Z., & Peter, Y. (2020, July). Personalized Feedbacks based on Learning Analytics to Enhance the Learning of Programming. In *IADIS 14th International Conference on e-Learning 2020*.

[52] Houssein, S. A., & Peter, Y. (2018, October). Evaluation of algorithms to support novice programmer. In *Proceedings of the 10th International Conference on Education Technology and Computers* (pp. 383-387).

[53] Howell, R., & Wong, S. H. S. (2018, December). Making the Most of Repetitive Mistakes: An Investigation into Heuristics for Selecting and Applying Feedback to Programming Coursework. In *2018 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)* (pp. 286-293). IEEE.

[54] Hu, Z., & Gehringer, E. F. (2019, October). Improving feedback on github pull requests: A bots approach. In 2019 IEEE Frontiers in Education Conference (FIE) (pp. 1-9). IEEE.

[55] Huang, S. B., Lai, C. F., & Jeng, Y. L. (2021). Real-Time Feedback Learning System Based on Programming Logs Analysis. *Journal of Internet Technology*, *22*(4), 779-787.

[56] Hubwieser, P., Berges, M., Striewe, M., & Goedicke, M. (2017, April). Towards competency based testing and feedback: Competency definition and measurement in the field of algorithms & data structures. In *2017 IEEE Global Engineering Education Conference (EDUCON)* (pp. 517-526). IEEE.

[57] Hwang, G. J., Liang, Z. Y., & Wang, H. Y. (2016, September). An Online Peer Assessment-Based Programming Approach to Improving Students' Programming Knowledge and Skills. In *2016 International Conference on Educational Innovation through Technology (EITT)* (pp. 81-85). IEEE.

[58] Jansen, J., Oprescu, A., & Bruntink, M. (2017). The impact of automated code quality feedback in programming education. In *Post-proceedings of the Tenth Seminar on Advanced Techniques and Tools for Software Evolution (SATToSE)* (Vol. 210).

[59] Jerše, G., & Lokar, M. (2018). Providing better feedback for students solving programming tasks using project tomo.

[60] Jiménez, S., Juárez-Ramírez, R., Castillo, V. H., Ramírez-Noriega, A., Márquez, B. Y., & Alanis, A. (2021). The Role of Personality in Motivation to use an Affective Feedback System. *Programming and Computer Software*, *47*(8), 793-802.

[61] Keuning, H., Jeuring, J., & Heeren, B. (2018). A systematic literature review of automated feedback generation for programming exercises. *ACM Transactions on Computing Education (TOCE)*, *19*(1), 1-43.

[62] Kitchenham, Barbara; Charters, Stuart. Guidelines for performing systematic literature reviews in software engineering. 2007.

[63] Krugel, J., Hubwieser, P., Goedicke, M., Striewe, M., Talbot, M., Olbricht, C., ... & Zettler, S. (2020, April). Automated measurement of competencies and generation of feedback in object-oriented programming courses. In *2020 IEEE Global Engineering Education Conference (EDUCON)* (pp. 329-338). IEEE.

[64] Krusche, S., & Seitz, A. (2018, February). Artemis: An automatic assessment management system for interactive learning. In Proceedings of the 49th ACM technical symposium on computer science education (pp. 284-289).

[65] Landa, R., & Martinez-Treviño, Y. (2019, June). Relevance of immediate feedback in an introduction to programming course. In *2019 ASEE Annual Conference & Exposition*.

[66] Latih, R., Bakar, M. A., Jailani, N., Ali, N. M., Salleh, S. M., & Zin, A. M. (2017, November). PC 2 to support instant feedback and good programming practice. In *2017 6th International Conference on Electrical Engineering and Informatics (ICEEI)* (pp. 1-5). IEEE.

[67] Le, D. M. (2022). Model-based automatic grading of object-oriented programming assignments. *Computer Applications in Engineering Education*, *30*(2), 435-457.

[68] Lee, H. H. (2021, March). Effectiveness of Real-time Feedback and Instructive Hints in Graduate CS Courses via Automated Grading System. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education* (pp. 101-107).

[69] Li, J., Ling, L., & Tan, C. W. (2021, June). Blending Peer Instruction with Just-In-Time Teaching: Jointly Optimal Task Scheduling with

Feedback for Classroom Flipping. In *Proceedings of the Eighth ACM Conference on Learning@ Scale* (pp. 117-126).

[70] Liénardy, S., Leduc, L., Verpoorten, D., & Donnet, B. (2020, February). CAFÉ: Automatic correction and feedback of programming challenges for a CS1 Course. In *Proceedings of the Twenty-Second Australasian Computing Education Conference* (pp. 95-104).

[71] Long III, L., & Morello, C. (2019). Using More Frequent and Formative Assessment When Replicating the Wright State Model for Engineering Mathematics Education.

[72] Makhlouf, J., & Mine, T. (2020, November). Automatic feedback models to students freely written comments. In *28th International Conference on Computers in Education, ICCE 2020* (pp. 336-341). Asia-Pacific Society for Computers in Education.

[73] Manso, A., Marques, C. G., & Santo, P. (2019, November). Teaching and Learning How to Programam with Algorithmi. In *2019 International Symposium on Computers in Education (SIIE)* (pp. 1-5). IEEE.

[74] Marwan, S. (2020, August). Investigating Best Practices in the Design of Automated Feedback to Improve Students' Performance and Learning. In *Proceedings of the 2020 ACM Conference on International Computing Education Research* (pp. 328-329).

[75] Marwan, S., Chi, M., Price, T. W., & Barnes, T. (2020). The Impact of Data-driven Positive Programming Feedback: When it Helps, What Happens when it Goes Wrong, and How Students Respond.

[76] Marwan, S., Gao, G., Fisk, S., Price, T. W., & Barnes, T. (2020, August). Adaptive immediate feedback can improve novice programming engagement and intention to persist in computer science. In *Proceedings of the 2020 ACM conference on international computing education research* (pp. 194-203).

[77] Marwan, S., Lytle, N., Williams, J. J., & Price, T. (2019, July). The impact of adding textual explanations to next-step hints in a novice programming environment. In Proceedings of the 2019 ACM conference on innovation and technology in computer science education (pp. 520-526).

[78] Marwan, S., Price, T. W., Chi, M., & Barnes, T. (2020). Immediate Data-Driven Positive Feedback Increases Engagement on Programming Homework for Novices. In *CSEDM@ EDM*.

[79] McBroom, J., Yacef, K., Koprinska, I., & Curran, J. R. (2018, June). A data-driven method for helping teachers improve feedback in computer programming automated tutors. In International Conference on Artificial Intelligence in Education (pp. 324-337). Springer, Cham.

[80] Mirmotahari, O., Berg, Y., Gjessing, S., Fremstad, E., & Damsa, C. (2019, April). A case-study of automated feedback assessment. In 2019 IEEE Global Engineering Education Conference (EDUCON) (pp. 1190-1197). IEEE.

[81] Mirza, D., Conrad, P. T., Lloyd, C., Matni, Z., & Gatin, A. (2019, July). Undergraduate teaching assistants in computer science: a systematic literature review. In *Proceedings of the 2019 ACM Conference on International Computing Education Research* (pp. 31-40).

[82] Mitra, R., & Chavan, P. (2019, March). DEBE feedback for large lecture classroom analytics. In *Proceedings of the 9th International Conference on Learning Analytics & Knowledge* (pp. 426-430).

[83] Morshed Fahid, F., Tian, X., Emerson, A., B. Wiggins, J., Bounajim, D., Smith, A., ... & Lester, J. (2021, June). Progression Trajectory-Based Student Modeling for Novice Block-Based Programming. In *Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization* (pp. 189-200).

[84] Murphy, S., Matthews-Dibbins, L., Maguire, C., & Shemmell, P. (2021). Automating Feedback for Computing Vivas and Presentations-A Journey. In *Computing Education Practice 2021* (pp. 1-4).

[85] Muuli, E., Lepp, M., Palm, R., & Luik, P. (2021, October). Automation of assessment and feedback in IT teaching from the teaching staff perspective. In *2021 IEEE Frontiers in Education Conference (FIE)* (pp. 1-9). IEEE.

[86] Nabil, R., Mohamed, N. E., Mahdy, A., Nader, K., Essam, S., & Eliwa, E. (2021, May). EvalSeer: An Intelligent Gamified System for Programming Assignments Assessment. In *2021 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC)* (pp. 235-242). IEEE.

[87] NARCISS, Susanne. Designing and evaluating tutoring feedback strategies for digital learning. Digital Education Review, n. 23, p. 7-26, 2013.

[88] Nees Jan Van Eck and Ludo Waltman. 2010. Software survey:

[89] Penmetsa, P. (2021). *Investigate Effectiveness of Code Features in Knowledge Tracing Task on Novice Programming Course*. North Carolina State University.

[90] Pettit, R. S., Homer, J., & Gee, R. (2017, March). Do Enhanced Compiler Error Messages Help Students? Results Inconclusive. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education* (pp. 465-470).

[91] Prisco, A., dos Santos, R., Nolibos, Á., Botelho, S., Tonin, N., & Bez, J. (2020, October). Evaluating a programming problem recommendation model-a classroom personalization experiment. In 2020 IEEE Frontiers in Education Conference (FIE) (pp. 1-6). IEEE.

[92] Hajja, Ayman, Austin J. Hunt, and Renée McCauley. "PolyPy: A Web-Platform for Generating Quasi-Random Python Code and Gaining Insights on Student Learning." 2019 IEEE Frontiers in Education Conference (FIE). IEEE, 2019.

[93] Qoiriah, A., Yamasari, Y., Nurhidayat, A. I., & Harimurti, R. (2020, December). Exploring Automatic Assessment-Based Features for Clustering of Students' Academic Performance. In International Conference on Soft Computing and Pattern Recognition (pp. 125-134). Springer, Cham.

[94] Bouvier, D., Lovellette, E., & Matta, J. (2021, February). Overnight Feedback Reduces Late Submissions on Programming Projects in CS1. In Australasian Computing Education Conference (pp. 176-180).

[95] Renzella, J., & Cain, A. (2020, October). Enriching programming student feedback with audio comments. In *2020 IEEE/ACM 42nd International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)* (pp. 173-183). IEEE.

[96] Rubinstein, A., Parzanchevski, N., & Tamarov, Y. (2019, July). In-Depth Feedback on Programming Assignments Using Pattern Recognition and Real-Time Hints. In Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education (pp. 243-244).

[97] Rump, A., Fehnker, A., & Mader, A. (2021, June). Automated assessment of learning objectives in programming assignments. In *International Conference on Intelligent Tutoring Systems* (pp. 299-309). Springer, Cham.

[98] Sana'a, M. A., Dousay, T. A., & Jeffery, C. L. (2020, October). Integrated Learning Development Environment for Learning and Teaching C/C++ Language to Novice Programmers. In *2020 IEEE Frontiers in Education Conference (FIE)* (pp. 1-5). IEEE.

[99] Seanosky, J., Guillot, I., Boulanger, D., Guillot, R., Guillot, C., Kumar, V., ... & Munshi, A. (2017, July). Real-time visual feedback: a study in coding analytics. In *2017 IEEE 17th International Conference on Advanced Learning Technologies (ICALT)* (pp. 264-266). IEEE.

[100] Serdyukova, Natalia A.; Serdyukov, Vladimir I.; NEUSTROEV, Sergey S. Testing as a feedback in a smart university and as a component of the identification of smart systems. In: Smart Education and e-Learning 2019. Springer, Singapore, 2019. p. 527-538.

[101] Silva, P., Costa, E., & Araújo, J. R. D. (2019, June). An adaptive approach to provide feedback for students in programming problem solving. In *International Conference on Intelligent Tutoring Systems* (pp. 14-23). Springer, Cham.

[102] Stephens-Martinez, K., & Fox, A. (2018, July). Giving hints is complicated: understanding the challenges of an automated hint system based on frequent wrong answers. In *Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education* (pp. 45-50).

[103] Sun, J. C. Y., & Hsu, K. Y. C. (2019). A smart eye-tracking feedback scaffolding approach to improving students' learning self-efficacy and performance in a C programming course. *Computers in Human Behavior*, *95*, 66-72.

[104] Teusner, R., Hille, T., & Staubitz, T. (2018, June). Effects of automated interventions in programming assignments: evidence from a field experiment. In *Proceedings of the Fifth Annual ACM Conference on Learning at Scale* (pp. 1-10).

[105] Tiam-Lee, T. J., & Sumi, K. (2018, June). Adaptive feedback based on student emotion in a system for programming practice. In International conference on intelligent tutoring systems (pp. 243-255). Springer, Cham.

[106] Treviño, Y. M., & Cavazos, M. R. L. (2018, October). Effects of immediate feedback using ICT in a CS1 course that implements Mastery Learning. In *2018 IEEE Frontiers in Education Conference (FIE)* (pp. 1-5). IEEE.

[107] Tsai, Y. S., Mello, R. F., Jovanović, J., & Gašević, D. (2021, April). Student appreciation of data-driven feedback: A pilot study on OnTask. In *LAK21: 11th International Learning Analytics and Knowledge Conference* (pp. 511-517).

[108] Van der Merwe, A., Kruger, H., & Du Toit, T. (2017). An early alert feedback system in learning.

[109] Vaz, Rafael Filipe Novôa; Nasser, Lilian. Um Estudo sobre o Feedback Formativo na Avaliação em Matemática e sua Conexão com a Atribuição de Notas. Bolema: Boletim de Educação Matemática, v. 35, p. 3-21, 2021.

[110] VOSviewer, a computer program for bibliometric mapping. scientometrics 84, 2 (2010), 523–538.

[111] Wang, W., Fraser, G., Barnes, T., Martens, C., & Price, T. (2021). Execution-Trace-Based Feature Engineering To Enable Formative Feedback on Visual, Interactive Programs. *feedback*, *32*(1), 2.

[112] Wilson, J., Ahrendt, C., Fudge, E. A., Raiche, A., Beard, G., & MacArthur, C. (2021). Elementary teachers' perceptions of automated feedback and automated scoring: Transforming the teaching and learning of writing using automated writing evaluation. *Computers & Education*, *168*, 104208.

[113] Wünsche, B. C., Huang, E., Shaw, L., Suselo, T., Leung, K. C., Dimalen, D., ... & Lobb, R. (2019, January). CodeRunnerGL-An interactive web-based tool for computer graphics teaching and assessment. In *2019 International Conference on Electronics, Information, and Communication (ICEIC)* (pp. 1-7). IEEE.

[114] Xiaochun, G., Yiwei, W., & Jingming, Z. (2017, December). The Self-assessment in E-learning and Personalized Feedback Education. In *Proceedings of the 2017 9th International Conference on Education Technology and Computers* (pp. 141-145).

[115] Yan, L., Hu, A., & Piech, C. (2019, February). Pensieve: Feedback on coding process for novices. In *Proceedings of the 50th acm technical symposium on computer science education* (pp. 253-259).

[116] Yan, Y. X., Wu, J. P., Nguyen, B. A., & Chen, H. M. (2020, February). The impact of iterative assessment system on programming learning

[125] *Engineering: Software Engineering Education and Training* (pp. 139-150).

[126] Figueiredo, J., & García-Peñalvo, F. (2021, October). A Tool Help for Introductory Programming Courses. In *Ninth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'21)* (pp. 18-24).

[127] Latih, R., Bakar, M. A., Jailani, N., Ali, N. M., Salleh, S. M., & Zin, A. M. (2017, November). PC 2 to support instant feedback and good programming practice. In 2017 6th International Conference on Electrical Engineering and Informatics (ICEEI) (pp. 1-5). IEEE.

behavior. In *Proceedings of the 2020 9th International Conference on Educational and Information Technology* (pp. 89-94).

[117] Yen, C. W., & Wang, T. I. (2017, July). Using self-explanation and ontology for providing proper feedbacks in a programming environment. In *2017 6th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI)* (pp. 585-590). IEEE.

[118] Yousuf, B., Conlan, O., & Wade, V. (2020, September). Assessing the Impact of the Combination of Self-directed Learning, Immediate Feedback and Visualizations on Student Engagement in Online Learning. In *European Conference on Technology Enhanced Learning* (pp. 274-287). Springer, Cham.

[119] Zampirolli, F. A., Borovina Josko, J. M., Venero, M. L., Kobayashi, G., Fraga, F. J., Goya, D., & Savegnago, H. R. (2021). An experience of automated assessment in a large-scale introduction programming course. *Computer Applications in Engineering Education*, *29*(5), 1284-1299.

[120] Zamprogno, L., Holmes, R., & Baniassad, E. (2020, November). Nudging student learning strategies using formative feedback in automatically graded assessments. In Proceedings of the 2020 ACM SIGPLAN Symposium on SPLASH-E (pp. 1-11).

[121] Zhi, R., Marwan, S., Dong, Y., Lytle, N., Price, T. W., & Barnes, T. (2019). Toward Data-Driven Example Feedback for Novice Programming. International Educational Data Mining Society.

[122] Gusukuma, Luke. "A Misconception Driven Student Model to Author Feedback." Proceedings of the 2018 ACM Conference on International Computing Education Research. 2018.

[123] Figueiredo, J., & García-Peñalvo, F. (2021, September). Teaching and Learning Tools for Introductory Programming in University Courses. In 2021 International Symposium on Computers in Education (SIIE) (pp. 1-6). IEEE.

[124] Ahmed, U. Z., Srivastava, N., Sindhgatta, R., & Karkare, A. (2020, June). Characterizing the pedagogical benefits of adaptive feedback for compilation errors by novice programmers. In *Proceedings of the ACM/IEEE 42nd International Conference on Software*