

Random and systematic errors in pairwise computer programming: A systematic review

Daniel Andrade Girón¹, Juana Sandivar Rosas², William Marín-Rodríguez^{1,3,*}, Edgardo Carreño Cisneros¹, Ernesto Diaz-Roncero¹, Henry Villarreal-Torres⁴

¹ Universidad Nacional José Faustino Sánchez Carrión. Huacho, Perú

² Universidad Nacional Mayor de San Marcos. Lima, Perú

³ Universidad Tecnológica del Perú. Lima, Perú

⁴ Universidad San Pedro. Chimbote, Perú

*Corresponding author: wmarin@unjfsc.edu.pe, C25859@utp.edu.pe

Abstract

In this article, a systematic review is carried out to identify random and systematic errors in studies on computer programming in pairs in higher education students. Methodologically, we applied the fundamentals of the PRISMA statement. One thousand one hundred eighty articles were selected from the Scopus, Web of Science, and IEEE Xplore databases. After a filtering process, the final sample was 23 23 articles. The results showed that couple programming has positive effects. The existence of both random and systematic errors was observed, which questions the internal and external validity. Further research is needed to establish the benefits of couple programming more precisely.

Keywords: Computer programming, higher education, artificial intelligence, big data.

Received on 17 February 2023, accepted on 10 May 2023, published on 23 May 2023

Copyright © 2023 Andrade Girón *et al.*, licensed to EAI. This is an open access article distributed under the terms of the [CC BY-NC-SA 4.0](https://creativecommons.org/licenses/by-nc-sa/4.0/), which permits copying, redistributing, remixing, transformation, and building upon the material in any medium so long as the original work is properly cited.

doi: 10.4108/eetsis.vi.3377

1. Introduction

The rapid changes in the labor market have generated a high demand for professionals with computer skills worldwide. Many higher education graduates need to be made aware of the skills the labor market requires. At the same time, some universities still need to prepare to offer teaching according to the needs of society.¹

The proliferation of big data and artificial intelligence (AI) technologies is transforming various aspects of our society, such as the economy, organizations, social relations, commercial transactions, education, and politics.^{2,3} AI has become an essential tool for solving problems and making decisions. Programming is a skill that allows you to make the most of the potential of AI and is, therefore, crucial for development and innovation.^{4,5}

Studies show that computer science and information systems students struggle to complete computer programming courses successfully. Learning to code is

complex, and failure and dropout rates in college-level programming classes remain high.⁶ There is a latent problem of low performance, student desertion, and demotivation in the face of computer programming in the careers in which related subjects are studied. This happens very frequently in the Engineering career.^{7,8} The overall probability of passing a first introductory programming course the first time has been 40% across all majors, with an initial failure rate of 19.5% and a dropout rate of 40.5%.⁹ These results are worrisome for higher education managers, as well as for programming language teachers. Many authors have proposed methodologies for teaching the programming language. They affirm that there is no perfect or unique methodology, but it should be able to cope with all learning styles. Each student has different knowledge, motivations, conditions, and abilities, so each has a different learning process.^{10,11}

Learning to program is a multifaceted process, combining theory and practice. This process implies acquiring skills in analyzing, designing, and implementing computer programs and developing efficient algorithms.¹²⁻¹⁴ For

many years -and even today-most introductory programming courses have focused on developing skills with educational models and assessment systems based on the individual paradigm. These use exams and programming tasks so that the student develops personalized skills and can finally program.¹⁵⁻¹⁷

However, software development is rarely a solitary activity. A software project comprises different phases, from analysis to evaluation. In each of them, a work team participates that shares methodologies, programming language paradigms, databases, a dictionary of terms used in the software, and methods for evaluating the software's quality. Therefore, working individually would be complex in an actual application project. Software development resembles a team sport in which working alone will affect quality, team morale, and the ability to overcome the complexity and risks associated with the project.¹⁸

Numerous scientific investigations have been carried out on pair programming.¹⁹⁻²³ Some researchers argue that pair programming is neither as economical nor productive as individual programming.²⁴ In comparison, others point out the need for more studies in this regard.²⁵⁻²⁸

The divergence in the results may be due to the investigation's validity. In explanatory and experimental research, validity is a determining factor. Therefore, special care must be taken with the instruments used to measure the applicability of the phenomenon. In all research, the minimization of errors must be guaranteed since there can be two types: random and systematic. Regarding the random error, the size and selection of the sample significantly influence validity.²⁹

Furthermore, this error is related to the concept of precision since an estimate or measurement is more precise the more minor the random error component is. In contrast, systematic error is due to factors such as the lack of control of extrinsic variables or the poor calibration of the measuring instruments. It is essential to identify and minimize random and systematic errors in an experimental study to guarantee the validity and reliability of the results obtained.^{30,31}

Systematic reviews on pair programming have analyzed different factors that influence student effectiveness. Such is the case of the study by Salleh (2008),³² who identified the factors that affect the effectiveness of students. This research focused on psychosocial factors such as compatibility, personality, and gender issues. The results showed that personality type is the most investigated factor in these studies. Subsequently, Salleh et al. (2011) conducted a meta-analysis of 74 studies to identify the factors that make pair programming highly effective. They reviewed articles published during the period from 1999 to 2007. Their study revealed that a student's skill level greatly affected the effectiveness of pair programming.^{33,34}

In a recent study, Satratzemi, Stelios, and Tsompanoudi (2022) conducted a systematic literature review (SLR) that included 57 studies on distributed pair programming (DPP) in higher education.³⁵ The objective of this review was to identify those studies that investigated factors related to the effectiveness of DPP as a method for learning to program,

as well as factors associated with mediating and stimulating interactions between students. In addition, the measures and instruments used to explore these factors and the tools and their characteristics were analyzed. Xu and Correia (2023) conducted a systematic review of DPP studies published after 2010 to understand the issues and factors that impact the effectiveness of the DPP team.³⁶ The results showed that individual characteristics such as previous programming experience, actual and perceived ability, gender, personality, time management, confidence, and self-esteem had been the subject of significant research.

Regardless of the investigations, studies focused on specifically analyzing random and systematic errors concerning the problem posed have yet to be found. This article aims to systematically review couple programming to identify and analyze random and systematic errors in the studies analyzed.

2. Methodology

This study has been developed using the PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) methodology for systematic reviews³⁷ based on the guidelines of Serrano, Navarro, & González (2022).³⁸ The following phases have been developed:

- First phase: formulating a clear and specific question that must be answered through the systematic review.
- Second phase: Conduct a comprehensive literature search to identify all relevant studies that address the research question.
- Third phase: application of inclusion and exclusion criteria to the identified studies to select those that meet the eligibility criteria for the systematic review.
- Fourth phase: evaluation of the methodological quality of the studies included in the systematic review using standardized tools.
- Fifth phase: extracting relevant data from each study included in the systematic review and performing statistical analysis and data synthesis.
- Sixth phase: interpreting the systematic review results based on the research question and identifying the practical implications and possible future research directions.
- Seventh phase: writing a detailed report of the systematic review. This report includes a complete description of the methodology used, the results obtained, and the review's conclusions.³⁹⁻⁴¹

Search strategy

An exhaustive search was carried out in specialized databases to find relevant information. Table 1 details the strategy used in the investigation.

Database	Search strategy
Scopus	<i>TITLE-ABS-KEY((effect OR influence) AND (collaboration OR cooperation OR</i>

	<i>association) AND (performance OR achievement) AND (programming OR "Pair Programming")</i>
IEEE Xplore Digital Library	<i>("Document Title": effect AND "Document Title":collaboration OR "Document Title":cooperation AND "Document Title":performance AND "Document Title":programming OR "Document Title":"Pair Programming") OR ("Abstract":influence) AND ("Abstract":association) AND ("Abstract":achievement)</i>
Web of Science	<i>((TI=(performance OR achievement)) AND ALL=(collaboration)) AND ALL=(programming)</i>

Table 1. Search strategy in the selected databases.

Inclusion and exclusion criteria

The inclusion and exclusion criteria refer to the standards and rules we previously established to determine which studies will be considered in the systematic review. These criteria are based on the research objectives and the research question addressed.

Characteristic	Inclusion	Exclusion
Participants	Higher education programming students	Basic education students
Fenómeno de interés	Pairwise programming	Individual programming
Time period	2002-2022	Studies outside this time interval
Context	Pair programming in higher education centers	Enterprise pairwise computer programming
Language	English	Non-English
Population and sample	Population and sample description	No description
Approach	Quantitative	Qualitative
Design	Experimental	Experimental

Table 2. Inclusion and exclusion criteria.

Sample selection

After processing the inclusion and exclusion criteria, the sample was restricted. This was done to analyze only those articles related to the proposed objective. In the flow diagram of Figure 1, 1180 identified articles are detailed. After eliminating duplication and applying the inclusion and exclusion criteria, 93 articles were obtained. Articles were excluded for different reasons, finally leaving a sample of 23.

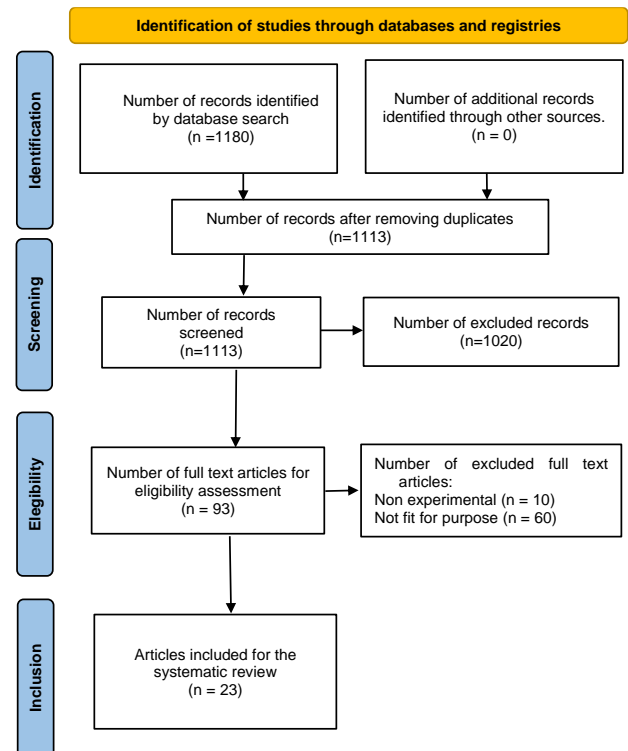


Figure 1. Flowchart of the search method and selection of references of the systematic review.

3. Results

The most relevant characteristics are found in Table 3. The attributes were considered: author, year, country, research design, sample, sample selection, type of test, control group, and test. The results obtained in this study show the distribution of the selected articles according to their country of origin. Most articles (47.83%) were published in the US, followed by Greece (21.74%). The other contributing countries were Turkey, Ireland, the United Arab Emirates, Germany, the Philippines, and Australia. Each one represented 4.35% of the total of the selected articles. These findings suggest great internationality in the subject studied, although it is essential to note that the database selection may have influenced this distribution.

Studies often use a quantitative approach and experimental design to address the research problem. In addition, 60.87% involve intervention and include a control group. This allows a more accurate and reliable comparison of the results obtained. At the same time, 39.13% of the studies did not consider a control group in their experimental designs.

34.78% used non-parametric tests, and 65.22% used parametric tests. Regarding the specific tests used, it was observed that 39.13% of the studies used the Student t-test, followed by Mann-Whitney U (17.39%), Wilcoxon (8.69%), and Factor Analysis, Kruskal Wallis and Chi-square with 4.35% each. 39.13% of the participants were recruited voluntarily, and 60.87% were students in programming courses.

Table 3. Descriptive characteristics of the included studies.

#	Author	Year	Country	Research design	Sample	Sample selection	Control group	Test type	Test -Test
01	Rodriguez J. F et al. ⁴²	2017	USA	Pre experimental intervention	54	Voluntaries	no	Nonparametric	Wilcoxon
02	Demir, O. ⁴³	2020	Turkey	Quasi-experimental intervention	64	Enrolled in subject	yes	Noparamétrica	U de Mann-Whitney
03	Isong, B et al. ²³	2016	USA	Quasi-experimental intervention	42	Enrolled in subject	yes	Parametric	t-Student
04	Beasley, Z et al. ⁴⁴	2022	Ireland	Quasi-experimental intervention	88	Enrolled in subject	yes	Parametric	t-Student
05	Salleh, N et al. ³²	2011	USA	Quasi-experimental intervention	218	Voluntaries	yes	Parametric	ANOVA
06	Domino, M et al. ⁴⁵	2007	USA	Quasi-experimental intervention	216	Voluntaries	yes	Nonparametric	Kruskal_Wallis
07	Geringer, E et al.	2003	USA	Quasi-experimental intervention	59	Voluntaries	no	Parametric	ANOVA
08	Ghobadi, S. ⁴⁶	2015	USA	Quasi-experimental intervention	64	Enrolled in subject	yes	Parametric	Análisis Factorial
09	Gold, C. ⁴⁷	2019	United Arab Emirates	Quasi-experimental intervention	71	Enrolled in subject	no	Nonparametric	U de Mann-Whitney
10	Zacharis, N. ⁴⁸	2007	Greece	Quasi-experimental intervention	129	Enrolled in subject	no	Parametric	t-Student
11	Tsompanoudi, D. ⁴⁹	2015	Greece	Quasi-experimental intervention	48	Voluntaries	yes	Parametric	t-Student
12	Kavitha, R. ⁵⁰	2013	USA	Quasi-experimental intervention	56	Enrolled in subject	yes	Nonparametric	Chi-Square
13	McDowell, C. ⁵¹	2002	USA	Quasi-experimental intervention	555	Enrolled in subject	no	Parametric	ANOVA
14	Sajev, A. ⁵²	2013	Germany	Quasi-experimental intervention	144	Enrolled in subject	yes	Parametric	t-Student
15	Sfetsos, P. ⁵³	2009	Greece	Quasi-experimental intervention	70	Enrolled in subject	yes	Parametric	t-Student
16	Sison, R. ⁵⁴	2009	Philippines	Experimental	48	Enrolled in subject	yes	Nonparametric	Wilcoxon
17	Xinogalos, S. ⁵⁵	2017	Greece	Pre experimental intervention	62	Enrolled in subject	no	Parametric	ANOVA
18	Zacharis, N. ⁵⁶	2010	Greece	Pre experimental intervention	64	Enrolled in subject	no	Parametric	t-Student
19	Smith, M et al. ⁵⁷	2017	USA	Pre experimental intervention	2468	Enrolled in subject	no	Parametric	ANOVA

20	Han, k et al. ⁵⁸	2010	USA	Quasi-experimental intervention	115	Enrolled in subject	yes	Parametric	t-Student
21	Vanhanen, J. ⁵⁹	2005	Australia	Pre experimental intervention	20	Voluntaries	no	Nonparametric	U de Mann-Whitney
22	Swamidurai, R, et al. ⁶⁰	2014	USA	Quasi-experimental intervention	10	Voluntaries	yes	Parametric	t-Student
23	Demir O, et al. ⁶¹	2021	Turkey	Quasi-experimental intervention	64	Voluntaries	yes	Nonparametric	U de Mann-Whitney

In the study by Salleh et al. (2011),³² an experimental investigation was carried out during the first semester of 2009. It used first-year undergraduate students enrolled in an introductory programming course. Of the 453 students enrolled in the COMPSCI 101 course, 317 responded to the demographic survey, and 218 (48%) completed the personality test. It is essential to highlight that the sample was not selected randomly, which implies a bias in the results obtained.

In the study by Adeliyi et al. (2021),¹⁹ two exploratory analyses were conducted to investigate the variables affecting pair programming. In the first study, 300 first- and second-year students participated; but only 16 enrolled. This is common in part-time distance learning contexts. In the second study, 1769 students were invited; but only 24 completed the final examination. In the same way, the sample was not selected randomly, implying a bias in the results obtained.

Saltz and Shamshurin (2017)⁶² evaluated pair programming in a graduate course on data science. One hundred ten students participated in it, divided into sections of 20 to 24 students. However, it is not specified whether the sample was selected in a representative manner, implying a possible limitation in the results. In the research by Omer and Suleyman (2021),⁶¹ the study group consisted of 64 volunteer junior and senior students from a university's Department of Information Technology Education and Educational Technology. This indicates a selection bias and affects the generalizability of the results. In Gehringer's (2003) study,⁶³ all students were surveyed at the end of the semester. Responses were only received from 59 of the 96 students who participated in the projects. This shows a limitation in the sample's representativeness due to the high rate of absent participants. While in the Tsompanoudi, Satratzemi, and Xinogalos' study (2015),⁴⁹ Computer Science students with basic knowledge of the Java programming language were voluntarily recruited. The sample consisted of 48 students, of which 10% were first year, 23% second year, 27% third year, 11% fourth year, and 29% fifth year. Although most participants had taken a Java course in the past, an obvious selection bias was observed due to the lack of uniformity in the sample. This also limits the generalizability of the results.

On the other hand, in the study by Sfetsos et al. (2009),⁵³ 70 students out of 90 were voluntarily selected to participate in the experiment. This shows a selection bias

that affects the representativeness of the results. It is important to note that in many studies on pair programming, an utterly random assignment of students is impossible due to the need to specifically pair students to maximize the chances that they will work together.⁶⁴ In addition, it is essential to specify the validation and reliability of the instruments used in most investigations. Random error is an important factor in research due to the variability inherent in measuring the variables and the characteristics of the voluntary participants, such as motivation, abilities, and competencies.³⁰

In many investigations, an error is made in the selection of the subjects since the nature of the study forces the selection of adequate samples for the research. Often this occurs for convenience, which is added to the voluntary participation of the students. An acceptable procedure is not always followed to determine the representative sample of students who will be part of the investigation. Regarding the design of the analytical studies, the aim is to estimate the effect of a study factor on a response variable. To guarantee an adequate inference, in addition to the group exposed to the study factor, it is necessary to use a control group to reference what happens in the subjects not exposed to said factor. This way, the results obtained in both groups can be compared, and more precise and reliable conclusions can be obtained. However, 39.13% have not considered a control group in the results received.

4. Discussion and conclusions

A systematic review of the research on pair programming in higher education was carried out, focusing on evaluating the validity of the experimental research.⁶⁵ When analyzing the reviewed articles, it was found that quantitative investigations of experimental design predominate, and only one investigation was oriented towards a mixed approach (qualitative and quantitative). Pair programming can have a positive impact on student learning. Bernadé and Liebenberg (2017)²¹ found that most students enjoy working in a team in programming, which is beneficial for them. Additionally, Saltz and Shamshurin (2017)⁶² reported that pair programming effectively improves communication and produces higher-quality code in less time for data scientists.⁶⁶ These findings suggest that this

programming may be a valuable strategy to improve student learning and performance in programming.

A study by Isong et al. (2016)²³ examined this technique in depth. It concluded that paired work is superior to individual programming regarding task completion time, correctness, and code quality. The null hypothesis was rejected, indicating that teamwork is significantly more effective than working alone. Beasley and Johnson (2022)⁴⁴ argue that pair programming offers a significant advantage in overall classroom performance compared to individual coding, even when pairs collaborate remotely. On the other hand, Salleh et al. (2011)³² have highlighted the influence of students' academic performance on pair programming, noting that high openness paired students perform better than their counterparts. In addition, Gehringer (2003) has shown that through pair programming and tools to track progress, students can avoid problems that often occur in team programming.⁶³ Likewise, Bernadé and Liebenberg (2017)²¹ reported that most students have a positive experience working in a team in programming. Saltz and Shamshurin (2017)⁶² found that pair programming can improve data scientists' communication and code quality. Additionally, Saltz and Shamshurin (2017)⁶² reported that pair programming effectively improves communication and produces higher-quality code in less time for data scientists. These findings suggest that this programming may be a valuable strategy to improve student learning and performance in programming.^{67,68}

The divergence in the results underlines the importance of analyzing the validity of the investigations. For this, it is necessary that the results can be generalized to a broader population, such as all programming students worldwide. Bennedsen and Caspersen (2007)⁶⁹ provide an idea of the size of this population, reporting that in 1999 there were more than one million students enrolled in computing in 72 countries. However, most of the research has used convenience samples and students who participated voluntarily, which is not representative of the general population. As a result, this creates bias and causes both random and systematic errors. Therefore, more research using random samples and more representative of the general population is needed to improve the external validity of the results.^{70,71}

After analyzing the studies included in our review, it was found that 39.13% correspond to pre-experimental studies. As is known, this type of research involves implementing an intervention or treatment; but without a control group to compare the results. Instead, a measurement is made before and after the intervention to determine if there was any change in the measured variable. Because it does not allow conclusively establishing a causal relationship, this type of study is considered the weakest regarding scientific evidence.⁷²

On the other hand, 60.87% of the studies included here have a quasi-experimental design. It is understood that in this type of research, variables are manipulated, but a random assignment of the participants to the groups is not carried out. Instead, groups can be formed by convenience,

specific characteristics of the participants, and geographic location, among others. The results are then compared between the groups to determine if the intervention had any effect. Despite not being as strong as a pure experimental study, this type of research is considered more robust than a pre-experimental study.⁷³ The results indicate that 46.66% of the investigations did not use a control group, meaning no reference group would allow comparing what happens in the groups not exposed to programming in pairs.⁷⁴ Therefore, without using a control group, it is difficult to determine if the results obtained are due to pair programming.

The investigations have not adequately analyzed the necessary procedures for all experimental research, such as the reliability and validity criteria of the measurement instruments and the conditions to carry out an empirical investigation. In all scientific research, it is essential to properly design the study to avoid errors that may compromise the proposed objectives. However, random errors have been found in most of the investigations, which can be attributed to the voluntary selection of participants and the investigator's convenience. This results in an unrepresentative sample. In addition, the inherent variability in the attribute measurement process can affect the validity and reliability of the assessment instruments. These factors can also influence the determination of the sample, the conformation of the pairs, and the selection of the control group. It has been shown that the lack of sensitivity in the application of the tests is one of the primary sources of error. Therefore, it is recommended to carry out investigations with more rigorous criteria in selecting the sample and applying the measurement instruments to guarantee the validity and reliability of the results obtained in the examination.

References

- [1]. Sibarani EM, Scerri S, Morales C, Auer S, Collarana D. Ontology-guided Job Market Demand Analysis: A Cross-Sectional Study for the Data Science field. Proceedings of the 13th International Conference on Semantic Systems, New York, NY, USA: Association for Computing Machinery; 2017, p. 25–32. <https://doi.org/10.1145/3132218.3132228>.
- [2]. Prüfer J, Prüfer P. Data science for entrepreneurship research: studying demand dynamics for entrepreneurial skills in the Netherlands. *Small Bus Econ* 2020;55:651–72. <https://doi.org/10.1007/s11187-019-00208-y>.
- [3]. Moreno MCC, Castro GLG. Unveiling Public Information in the Metaverse and AI Era: Challenges and Opportunities. *Metaverse Basic and Applied Research* 2023;2:35. <https://doi.org/10.56294/mr202335>.

- [4]. Lali K, Chakor A. Improving the Security and Reliability of a Quality Marketing Information System: A Priority Prerequisite for Good Strategic Management of a Successful Entrepreneurial Project. *Data & Metadata* 2023;2:40. <https://doi.org/10.56294/dm202340>.
- [5]. Verdesoto GJZ, Soto IBR, Alfaro AC. Contributions of neurosciences, neuromarketing and learning processes in innovation. *Salud, Ciencia y Tecnología* 2023;3:396–396. <https://doi.org/10.56294/saludcyt2023396>.
- [6]. Woszczyński AB, Guthrie TC, Shade S. Personality and programming. *Journal of Information Systems Education* 2005;16:293.
- [7]. Tobón AB. Diagnóstico de los estudiantes de ingeniería en diseño de entretenimiento digital para facilitar el aprendizaje de los fundamentos de programación. *Actas de Diseño* 2021.
- [8]. Gonzalez-Argote D. Thematic Specialization of Institutions with Academic Programs in the Field of Data Science. *Data & Metadata* 2023;2:24–24. <https://doi.org/10.56294/dm202324>.
- [9]. Watson C, Li FWB. Failure rates in introductory programming revisited. *Proceedings of the 2014 conference on Innovation & technology in computer science education*, New York, NY, USA: Association for Computing Machinery; 2014, p. 39–44. <https://doi.org/10.1145/2591708.2591749>.
- [10]. Compañ-Rosique P, Molina-Carmona R, Satorre-Cuerda R. Effects of Teaching Methodology on the Students' Academic Performance in an Introductory Course of Programming. In: Zaphiris P, Ioannou A, editors. *Learning and Collaboration Technologies. Designing Learning Experiences*, Cham: Springer International Publishing; 2019, p. 332–45. https://doi.org/10.1007/978-3-030-21814-0_25.
- [11]. Gonzalez-Argote D. Immersive environments, Metaverse and the key challenges in programming. *Metaverse Basic and Applied Research* 2022;1:6. <https://doi.org/10.56294/mr20226>.
- [12]. Tiwari P, Chaudhary S, Majhi D, Mukherjee B. Comparing research trends through author-provided keywords with machine extracted terms: A ML algorithm approach using publications data on neurological disorders. *Iberoamerican Journal of Science Measurement and Communication* 2023;3. <https://doi.org/10.47909/ijsmc.36>.
- [13]. Miceli JE, Castro M, Cordova DD. When links build networks: brief history about the Antropocaos Group. *AWARI* 2020;1:e013–e013. <https://doi.org/10.47909/awari.61>.
- [14]. Zaina RZ, Ramos VFC, Araujo GM de. Automated triage of financial intelligence reports. *Advanced Notes in Information Science* 2022;2:24–33. <https://doi.org/10.47909/anis.978-9916-9760-3-6.115>.
- [15]. Martínez R. Tipos de Métricas de calidad para validar datasets gubernamentales Argentinos. *Revista Abierta de Informática Aplicada* 2022;6:40–53. <https://doi.org/10.59471/raia20224>.
- [16]. Contreras DEÁ, Pérez CMD, Morales RH. Factores académicos asociados al proceso de investigación formativa en las instituciones educativas del sector oficial de Sincelejo, Sucre. *Región Científica* 2023;2:202319–202319. <https://doi.org/10.58763/rc202319>.
- [17]. Sánchez RM. Transformando la educación online: el impacto de la gamificación en la formación del profesorado en un entorno universitario. *Metaverse Basic and Applied Research* 2023;2:47. <https://doi.org/10.56294/mr202347>.
- [18]. Fitzpatrick B, Collins-Sussman B. *Team Geek: A Software Developer's Guide to Working Well with Others*. O'Reilly Media, Inc.; 2012.
- [19]. Adeliyi A, Wermelinger M, Kear K, Rosewell J. Investigating Remote Pair Programming In Part-Time Distance Education. *Proceedings of the 2021 Conference on United Kingdom & Ireland Computing Education Research*, New York, NY, USA: Association for Computing Machinery; 2021, p. 1–7. <https://doi.org/10.1145/3481282.3481290>.
- [20]. Andrade-Girón D, Carreño-Cisneros E, Mejía-Dominguez C, Marín-Rodríguez W, Villarreal-Torres H. Comparison of Machine Learning Algorithms for Predicting Patients with Suspected COVID-19. *Salud, Ciencia y Tecnología* 2023;3:336–336. <https://doi.org/10.56294/saludcyt2023336>.
- [21]. Benadé T, Liebenberg J. Pair Programming as a Learning Method Beyond the Context of Programming. *Proceedings of the 6th Computer Science Education Research Conference*, New York, NY, USA: Association for Computing Machinery; 2017, p. 48–55. <https://doi.org/10.1145/3162087.3162098>.
- [22]. Denner J, Werner L, Campe S, Ortiz E. Pair Programming: Under What Conditions Is It Advantageous for Middle School Students? *Journal of Research on Technology in*

- Education 2014;46:277–96. <https://doi.org/10.1080/15391523.2014.888272>.
- [23]. Isong B, Moemi T, Dladlu N, Motlhabane N, Ifeoma O, Gasela N. Empirical Confirmation of Pair Programming Effectiveness in the Teaching of Computer Programming. 2016 International Conference on Computational Science and Computational Intelligence (CSCI), 2016, p. 276–81. <https://doi.org/10.1109/CSCI.2016.0060>.
- [24]. Nawrocki J, Wojciechowski A. Experimental evaluation of pair programming. European Software Control and Metrics (Escom) 2001:99–101.
- [25]. Vanoy RJA. STEM Education as a Teaching Method for the Development of XXI Century Competencies. Metaverse Basic and Applied Research 2022;1:21. <https://doi.org/10.56294/mr202221>.
- [26]. Parrish A, Smith R, Hale D, Hale J. A field study of developer pairs: productivity impacts and implications. IEEE Software 2004;21:76–9. <https://doi.org/10.1109/MS.2004.1331306>.
- [27]. Arévalo YB, García MB. Scientific production on dialogical pedagogy: a bibliometric analysis. Data & Metadata 2023;2:7. <https://doi.org/10.56294/dm20237>.
- [28]. Soto IBR, Marín-Rodríguez WJ, Baldeos-Ardán LA, Lío-Jordán F de M, Villanueva-Cadenas DI, Soledispa-Cañarte BJ, et al. Teacher training, work, profession in the pandemic and post-pandemic context. Salud, Ciencia y Tecnología 2023;3:338–338. <https://doi.org/10.56294/saludcyt2023338>.
- [29]. Júnior EM da S, Dutra ML. A roadmap toward the automatic composition of systematic literature reviews. Iberoamerican Journal of Science Measurement and Communication 2021;1:1–22. <https://doi.org/10.47909/ijsmc.52>.
- [30]. Pallás JMA, Villa JJ. Métodos de investigación clínica y epidemiológica. Elsevier España; 2004.
- [31]. Machuca-Contreras F, Canova-Barrios C, Castro MF. Una aproximación a los conceptos de innovación radical, incremental y disruptiva en las organizaciones. Región Científica 2023;2:202324–202324. <https://doi.org/10.58763/rc202324>.
- [32]. Salleh N, Mendes E, Grundy J. The effects of openness to experience on pair programming in a higher education context. 2011 24th IEEE-CS Conference on Software Engineering Education and Training (CSEE&T), 2011, p. 149–58. <https://doi.org/10.1109/CSEET.2011.5876082>.
- [33]. Salleh N, Mendes E, Grundy J, Burch GStJ. An empirical study of the effects of personality in pair programming using the five-factor model. 2009 3rd International Symposium on Empirical Software Engineering and Measurement, 2009, p. 214–25. <https://doi.org/10.1109/ESEM.2009.5315997>.
- [34]. Cano CAG, Castillo VS, Gallego TAC. Mapping the Landscape of Netnographic Research: A Bibliometric Study of Social Interactions and Digital Culture. Data & Metadata 2023;2:25. <https://doi.org/10.56294/dm202325>.
- [35]. Satratzemi M, Stelios X, Tsompanoudi D. Distributed Pair Programming in Higher Education: A Systematic Literature Review. Journal of Educational Computing Research 2023;61:546–77. <https://doi.org/10.1177/07356331221122884>.
- [36]. Xu F, Correia A-P. Adopting distributed pair programming as an effective team learning activity: a systematic review. J Comput High Educ 2023. <https://doi.org/10.1007/s12528-023-09356-3>.
- [37]. Sánchez Meca J. Cómo realizar una revisión sistemática y un meta-análisis. Aula abierta 2010.
- [38]. Sánchez Serrano S, Pedraza Navarro I, Donoso González M. ¿Cómo hacer una revisión sistemática siguiendo el protocolo PRISMA?: Usos y estrategias fundamentales para su aplicación en el ámbito educativo a través de un caso práctico. Bordón: Revista de pedagogía 2022;74:51–66.
- [39]. Alexander PA. Methodological Guidance Paper: The Art and Science of Quality Systematic Reviews. Review of Educational Research 2020;90:6–23. <https://doi.org/10.3102/0034654319854352>.
- [40]. Pigott TD, Polanin JR. Methodological Guidance Paper: High-Quality Meta-Analysis in a Systematic Review. Review of Educational Research 2020;90:24–46. <https://doi.org/10.3102/0034654319877153>.
- [41]. Stern C, Lizarondo L, Carrier J, Godfrey C, Rieger K, Salmund S, et al. Methodological guidance for the conduct of mixed methods systematic reviews. JBI Evidence Synthesis 2020;18:2108. <https://doi.org/10.11124/JBISRIR-D-19-00169>.
- [42]. Rodríguez FJ, Price KM, Boyer KE. Exploring the Pair Programming Process: Characteristics of Effective Collaboration.

- Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education, New York, NY, USA: Association for Computing Machinery; 2017, p. 507–12. <https://doi.org/10.1145/3017680.3017748>.
- [43]. Demir E, Bilgin MH, Karabulut G, Doker AC. The relationship between cryptocurrencies and COVID-19 pandemic. *Eurasian Econ Rev* 2020;10:349–60. <https://doi.org/10.1007/s40822-020-00154-1>.
- [44]. Beasley ZJ, Johnson AR. The Impact of Remote Pair Programming in an Upper-Level CS Course. Proceedings of the 27th ACM Conference on on Innovation and Technology in Computer Science Education Vol. 1, New York, NY, USA: Association for Computing Machinery; 2022, p. 235–40. <https://doi.org/10.1145/3502718.3524772>.
- [45]. Domino MA, Collins RW, Hevner AR. Controlled experimentation on adaptations of pair programming. *Inf Technol Manage* 2007;8:297–312. <https://doi.org/10.1007/s10799-007-0016-8>.
- [46]. Ghobadi S, Campbell J, Clegg S. Pair programming teams and high-quality knowledge sharing: A comparative study of cooperative reward structures. *Inf Syst Front* 2017;19:397–409. <https://doi.org/10.1007/s10796-015-9603-0>.
- [47]. Gold-Veerkamp C, Klopp M, Abke J. Pair Programming as a Didactical Approach in Higher Education and its Evaluation. 2019 IEEE Global Engineering Education Conference (EDUCON), 2019, p. 1055–62. <https://doi.org/10.1109/EDUCON.2019.8725150>.
- [48]. Zacharis N. Evaluating the Effects of Virtual Pair Programming on Students' Achievement and Satisfaction. *International Journal of Emerging Technologies in Learning (IJET)* 2009;4:34–9.
- [49]. Tsompanoudi D, Satratzemi M, Xinogalos S. Distributed pair programming using collaboration scripts: An educational system and initial results. *Informatics in Education* 2015;14:291–314.
- [50]. Kavitha RK, Ahmed MSI. Knowledge sharing through pair programming in learning environments: An empirical study. *Educ Inf Technol* 2015;20:319–33. <https://doi.org/10.1007/s10639-013-9285-5>.
- [51]. McDowell C, Werner L, Bullock H, Fernald J. The effects of pair-programming on performance in an introductory programming course. Proceedings of the 33rd SIGCSE technical symposium on Computer science education, New York, NY, USA: Association for Computing Machinery; 2002, p. 38–42. <https://doi.org/10.1145/563340.563353>.
- [52]. Ummadi S, Shravani B, Rao NR, Reddy MS, Sanjeev B. Overview on controlled release dosage form. *System* 2013;7:51–60.
- [53]. Sfetsos P, Stamelos I, Angelis L, Deligiannis I. An experimental investigation of personality types impact on pair effectiveness in pair programming. *Empir Software Eng* 2009;14:187–226. <https://doi.org/10.1007/s10664-008-9093-5>.
- [54]. Sison R. Investigating the Effect of Pair Programming and Software Size on Software Quality and Programmer Productivity. 2009 16th Asia-Pacific Software Engineering Conference, 2009, p. 187–93. <https://doi.org/10.1109/APSEC.2009.71>.
- [55]. Xinogalos S, Satratzemi M, Chatzigeorgiou A, Tsompanoudi D. Student perceptions on the benefits and shortcomings of distributed pair programming assignments. 2017 IEEE Global Engineering Education Conference (EDUCON), 2017, p. 1513–21. <https://doi.org/10.1109/EDUCON.2017.7943050>.
- [56]. Zacharis NZ. Measuring the Effects of Virtual Pair Programming in an Introductory Programming Java Course. *IEEE Transactions on Education* 2011;54:168–70. <https://doi.org/10.1109/TE.2010.2048328>.
- [57]. Smith MO, Giugliano A, DeOrio A. Long Term Effects of Pair Programming. *IEEE Transactions on Education* 2018;61:187–94. <https://doi.org/10.1109/TE.2017.2773024>.
- [58]. Han K-W, Lee E, Lee Y. The Impact of a Peer-Learning Agent Based on Pair Programming in a Programming Course. *IEEE Transactions on Education* 2010;53:318–27. <https://doi.org/10.1109/TE.2009.2019121>.
- [59]. Vanhanen J, Lassenius C. Effects of pair programming at the development team level: an experiment. 2005 International Symposium on Empirical Software Engineering, 2005., 2005, p. 10 pp.-. <https://doi.org/10.1109/ISESE.2005.1541842>.
- [60]. Swamidurai R, Dennis B, Kannan U. Investigating the impact of peer code review and pair programming on test-driven development. *IEEE SOUTHEASTCON* 2014, 2014, p. 1–5. <https://doi.org/10.1109/SECON.2014.6950664>.
- [61]. Demir Ö, Seferoglu SS. The Effect of Determining Pair Programming Groups According to Various Individual Difference Variables on Group Compatibility, Flow, and Coding Performance. *Journal of Educational*

- Computing Research 2021;59:41–70. <https://doi.org/10.1177/0735633120949787>.
- [62]. Saltz JS, Shamshurin I. Does pair programming work in a data science context? An initial case study. 2017 IEEE International Conference on Big Data (Big Data), 2017, p. 2348–54. <https://doi.org/10.1109/BigData.2017.8258189>.
- [63]. Gehringer EF. A pair-programming experiment in a non-programming course. Companion of the 18th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications, New York, NY, USA: Association for Computing Machinery; 2003, p. 187–90. <https://doi.org/10.1145/949344.949397>.
- [64]. Williams L, McCrickard DS, Layman L, Hussein K. Eleven Guidelines for Implementing Pair Programming in the Classroom. Agile 2008 Conference, 2008, p. 445–52. <https://doi.org/10.1109/Agile.2008.12>.
- [65]. Mascarenhas HAD, Dias TMR. Academic education data as a source for analysis of the migration process for training. *Advanced Notes in Information Science* 2022;2:53–62. <https://doi.org/10.47909/anis.978-9916-9760-3-6.91>.
- [66]. Subbarayan S, Gunaseelan HG. A Review of Data and Document Clustering pertaining to various Distance Measures. *Salud, Ciencia y Tecnología* 2022;2:194–194. <https://doi.org/10.56294/saludcyt2022194>.
- [67]. Mascarenhas HAD, Dias TMR, Dias PM. Adoption of Network Analysis Techniques to Understand the Training Process in Brazil. *AWARI* 2020;1:e004–e004. <https://doi.org/10.47909/awari.63>.
- [68]. Martins DL. Data science teaching and learning models: focus on the Information Science area. *Advanced Notes in Information Science* 2022;2:140–8. <https://doi.org/10.47909/anis.978-9916-9760-3-6.100>.
- [69]. Bennedsen J, Caspersen ME. Failure rates in introductory programming. *SIGCSE Bull* 2007;39:32–6. <https://doi.org/10.1145/1272848.1272879>.
- [70]. Takaki P, Dutra M. Data science in education: interdisciplinary contributions. *Advanced Notes in Information Science* 2022;2:149–60. <https://doi.org/10.47909/anis.978-9916-9760-3-6.94>.
- [71]. Espinoza ELO, Ayala ACN, Yovera SER y, Soto FGC, Vilela AJM. Design Thinking como herramienta para fomentar la innovación y el emprendimiento. *Salud, Ciencia y Tecnología* 2023;3:368–368. <https://doi.org/10.56294/saludcyt2023368>.
- [72]. Campbell DT, Stanley JC. *Experimental and Quasi-Experimental Designs for Research*. Ravenio Books; 2015.
- [73]. Fernández P, Vallejo G, Livacic-Rojas P, Tuero E. Validez Estructurada para una investigación cuasi-experimental de calidad. Se cumplen 50 años de la presentación en sociedad de los diseños cuasi-experimentales. *Anales de Psicología / Annals of Psychology* 2014;30:756–71. <https://doi.org/10.6018/analesps.30.2.166911>.
- [74]. Larrosa JMC, Galgano F, Gutiérrez E. Kinship network evolution in Argentina. An exploration based on online data. *AWARI* 2022;3. <https://doi.org/10.47909/awari.150>.