**Abstract**

**Introduction**

The domain of artificial intelligence has reached a critical juncture with the advent of advanced generative AI systems such as ChatGPT. ChatGPT has undergone extensive training using a vast and varied corpus of textual data obtained from various sources on the internet (Borji, 2023; Ray, 2023). Through the process of extracting patterns and structures from a vast amount of textual data, these models have developed a notable ability to generate coherent and contextually appropriate responses to prompts in natural language. Even so, it is important to recognize that despite their high level of intelligence, these systems do not possess actual cognition or intentionality. The responses provided by the model are mostly derived from the identification and extrapolation of statistical patterns within the data it has been trained on.

The social effects surrounding the implementation of AI systems, which possess significant capabilities but are constrained in certain aspects, continue to be a subject of intense discussion and controversy (Wach et al., 2023). While accepting the limitations of a certain entity or concept, it is imperative to equally acknowledge its significant usefulness within appropriate boundaries. Domains like as customer service (Subagja, Ausat, Sari, Wanof, & Suherlan, 2023), creative writing assistance (Shidiq, 2023), and programming education (Tian et al., 2023) exemplify areas in which these technologies can enhance human talents and increase productivity. However, the emergence of sophisticated generative artificial intelligence systems like ChatGPT has generated significant attention on their capacity to revolutionize educational methodologies in various fields (Lee, 2023; Lo, 2023; Rahman & Watanobe, 2023), including the teaching of computer programming (Biswas, 2023; Surameery & Shakor, 2023). Nevertheless, despite the considerable attention and speculations around these emerging technologies, there is a scarcity of significant academic study investigating the effectiveness of tools such as ChatGPT in reliably teaching fundamental coding abilities without human assistance. The objective of this study is to fill the existing research gap by conducting a comprehensive examination of ChatGPT's effectiveness as a distinct programming teacher for beginner students who are learning the basics of Python.

This study aims to investigate the effectiveness of ChatGPT in facilitating independent learning of expertise in the Python programming language among students without prior knowledge, without the need for human instruction. This study examines the accuracy and standard of ChatGPT's explanations regarding fundamental concepts as well as its ability to provide informative sample code. In addition, this study proposes the pedagogical strategy employed by ChatGPT in comparison to traditional human tutoring methods, utilizing expert surveys as a way of evaluation. This study presents and assesses a well-defined conceptual framework that integrates both bottom-up and top-down approaches to facilitate independent learning.

The key findings indicate that ChatGPT offers excellent assistance in regards to fundamental Python concepts. However, it exhibits limitations when it comes to fostering the development of more advanced skills. Experts recommend integrating human tutoring to maximize outcomes. This research generates novel insights into ChatGPT's promise and pitfalls as an independent programming tutor, establishing an empirical foundation to guide the thoughtful and strategic incorporation of generative AI in computer science education.

**Literature Review**

The process of learning computer programming languages has been extensively studied within education and cognitive science research. An enduring observation evident in a multitude of research is that learning and successful teaching of programming pose significant challenges (Medeiros, Ramalho, & Falcão, 2018; Seng & Yatim, 2014). Research findings also indicate that students have challenges in the areas of reading (McCracken et al., 2001), writing (Lister et al., 2004), and programming code design (Tenenberg & Fincher, 2005). This challenge arises from the diverse array of advanced skills required, including critical thinking and problem-solving abilities, as well as proficiency in computer programming syntax and practical expertise in debugging and implementing solutions.

Several approaches have been suggested and assessed in order to facilitate the early phases of acquiring programming skills. Effective pedagogical tactics include the utilization of visual block-based languages as a precursor to shifting towards text-based coding (Lin & Weintrop, 2021), the implementation of problem-based learning using real-world illustrations (Martins, de Almeida Souza Concilio, & de Paiva Guimarães, 2018), and the provision of scaffolding to facilitate the development of planning and tracing abilities (Whalley & Kasto, 2014). The significance of imparting both theoretical foundations and practical coding skills is emphasized by the research community (Ismail, Ngah, & Umar, 2010). Research has also demonstrated that the utilization of pair programming and interactive instructors can effectively enhance engagement and foster skill development among those who are new to a particular domain (Isong, 2014).

The evolution of programming language learning has paralleled technical improvements, wherein generative AI tools such as ChatGPT have emerged as promising facilitators in this field. Several studies have been conducted to investigate the potential of utilizing ChatGPT in the realm of programming instruction within an experimental framework (Biswas, 2023). When utilizing AI-based tools and settings for programming learning, students possess the capacity to submit problems to the AI tool, thereby obtaining prompt feedback and solutions. This enables a customized educational experience that is in accordance with the student's unique learning pace (Yılmaz & Yılmaz, 2022). AI-powered tools have the potential to assist students with coding tasks through the provision of ideas, error detection, and the automation of code development. This approach has the potential to enhance students' ability to produce code that is both efficient and precise, hence minimizing the time and effort needed to fulfill programming tasks. The utilization of AI-driven tools and environments has the potential to enhance student engagement and motivation through interactive interaction and individualized assistance and feedback in the process of acquiring programming skills (Yilmaz & Yilmaz, 2023).

So, the investigation of self-directed learning of programming fundamentals exclusively through AI tools such as ChatGPT, without any human guidance, presents a promising domain for research exploration. The objective of this study is to investigate the potential of ChatGPT in facilitating the independent acquisition of fundamental ideas in a specific programming language among novice students. The focus of this study revolves around basic research questions:

1. Can ChatGPT effectively teach beginners a programming language without human help?
2. How accurately can ChatGPT explain programming concepts and provide sample code to novice learners compared to human tutors?
3. How do experts view the quality of programming guidance provided by ChatGPT compared to traditional methods?

The primary emphasis of our study lies in the utilization of ChatGPT as a conversational agent, specifically in the context of providing programming education through text-based instruction. This study aims to analyze the sequential curriculum offered by ChatGPT, focusing on its suitability for novice learners seeking to acquire basic computer programming knowledge. The accuracy, thoroughness, and instructional design quality of each module and explanation provided by ChatGPT will be subject to rigorous evaluation. Upon further examination, our analysis thoroughly dissects the sequential phases of programming education as outlined by ChatGPT. Next, we proceed to employ the litmus test by seeking the perspectives of proficient professionals who possess a deep understanding of the complexities of the specific programming language in question. This will be accomplished by the administration of a survey. These insights hold significant value as they contribute to the understanding of the effectiveness of ChatGPT's assistance.

**Methodology**

**Figure: Method Diagram**

**Programming Language Learning Approaches**

There are two distinct methodologies for acquiring proficiency in a programming language: the bottom-up approach (BUA) and the top-down approach (TDA).

**Bottom-Up Approach**

The bottom-up technique prioritizes the acquisition of foundational elements in a programming language, such as variables, data types, functions, and grammar. Through a comprehensive grasp of these fundamental principles, students can progressively integrate them to cultivate their abilities in a methodical manner, so establishing a solid technical foundation. This approach is often utilized in educational institutions and programming books targeted at novices (Saito & Yamaura, 2013).

**Top-Down Approach**

The top-down approach prioritizes an instructional approach in which learners are first exposed to sample programming in order to gain an understanding of language functionality (Saito & Yamaura, 2014). The utilization of TDA facilitates the process of language acquisition, requiring a shorter duration compared to BUA. This instructional approach enables learners to construct practical projects at an accelerated pace, resulting in heightened levels of engagement. Learners are provided with structured curriculum or specific learning outcomes, emphasizing comprehension, manipulation, and validation of code. This methodology promotes a comprehensive understanding of programming blocks as opposed to individual code portions. Nevertheless, it is possible that the initial stages of learning may exhibit a deficiency in fundamental principles. The top-down method places emphasis on comprehensive conceptualization and practical implementation, while initially de-emphasizing technical complexities.

**Approach Used by ChatGPT**

ChatGPT effectively integrates both bottom-up and top-down methodologies inside its instructional framework for teaching programming languages to novice learners. ChatGPT employs a bottom-up approach by providing fundamental explanations and details to establish a foundational understanding of programming concepts. Concurrently, it utilizes a top-down methodology by providing learners with concrete coding situations that allow them to comprehend the operation of the language inside the framework of real-life applications. The utilization of this dual approach provides learners with a complete learning experience. Through systematic explanations, learners get a strong foundation in programming principles. Simultaneously, they engage in practical code implementation, allowing for a hands-on learning experience. The combination of bottom-up and top-down strategies guarantees that learners acquire a comprehensive understanding of syntax and structure, as well as the capacity to create functional code. This approach finally cultivates a comprehensive expertise in programming languages.

**Framework for Learning Programming Using ChatGPT**

To facilitate learning programming by using ChatGPT we propose a conceptual framework. The proposed framework primarily targets individuals who are new to programming, but it is also adaptable to accommodate those with varying levels of programming expertise. While originally designed for beginners in programming, this framework has the potential to serve as a viable technique to learners of all proficiency levels who possess a strong inclination to go on a path towards programming knowledge. This is achieved by utilizing the assistance of ChatGPT as a guiding tool, without the need for human guidance. The flexibility of the framework promotes a dynamic and inclusive learning environment, making it a potential asset for anybody interested in programming, regardless of their existing knowledge in the field.

**Figure: Framework**

The following section provides a brief overview of the objectives included in the framework.

1. Introduction to ChatGPT Learning Platform:

Familiarize students with the ChatGPT platform as a self-learning resource for programming.

1. Selecting the Targeted Programming Language:

Students choose the specific programming language they want to learn.

1. Initial Exposure to the Language:

ChatGPT provides an overview of the selected programming language, introducing basic concepts and terminology.

1. Interactive Q&A Sessions:

Students engage in interactive sessions with ChatGPT, asking questions related to programming language fundamentals.

1. Bottom-Up Learning (BUA) Phase:

ChatGPT explains foundational concepts, syntax, and data structures, assisting students in building a strong base.

1. Top-Down Learning (TDA) Phase:

ChatGPT presents sample coding scenarios, allowing students to observe practical implementations of language concepts.

1. Hands-On Coding Practice:

Students apply their learning by actively writing code with ChatGPT's guidance.

1. Real-Time Feedback and Corrections:

ChatGPT provides instant feedback on code accuracy and helps correct errors, encouraging iterative learning.

1. Evaluation and Self-Assessment:

Students evaluate their understanding and coding skills by solving coding challenges independently.

The suggested framework incorporates the combination of two pedagogical approaches in programming education: a foundational approach that prioritizes the understanding of fundamental concepts, and a practical approach that stresses the implementation of these concepts in real-world scenarios. The utilization of ChatGPT as an instructional tool is aimed at offering help to individuals who are new to the programming. This approach aims to tackle two significant questions. Can ChatGPT provide effective programming instruction to beginner individuals without any human intervention? Moreover, can ChatGPT's explanations and code examples attain a level of quality comparable to those offered by human tutors?

The initial question is to the evaluation of ChatGPT's ability to operate independently as a proficient instructor. Our research aims to examine the effectiveness of gaining programming abilities exclusively through the use of ChatGPT, without any human instructor involvement. To determine the necessary steps, we employed ChatGPT to provide a comprehensive guide for new programmers who want to learn in programming with Python, covering from introductory to fundamental concepts. Subsequently, we assessed the generated responses by incorporating expert perspectives.

The following inquiry concerns the contrast between the instructional approaches utilized by ChatGPT and those employed by human educators. The primary objective of this study is to examine how ChatGPT effectively explains programming concepts and offers instructive code snippets. Subsequently, we will proceed to conduct a comparative analysis between this strategy and the instructional approaches utilized by human teachers. This facilitates the evaluation of the relative effectiveness of ChatGPT's guidance compared to human performance.

**Using ChatGPT for Learning Python**

In order to evaluate the ability of ChatGPT to offer concise and organized instructional assistance for individuals who are new to programming languages, we ask the following questions. Python was selected as the programming language for our research project. The utilization of a syntactically-simple programming language, such as Python, compared to a complex language like Java, has proven to enhance students' understanding and learning of programming principles (Koulouri, Lauria, & Macredie, 2014). Initially, we ask ChatGPT's guidance in obtaining a systematic plan to pursue learning proficiency in Python programming as an individual with limited experience in the field. Subsequently, we request the system to give instruction on diverse Python modules encompassing topics such as variables and data types, data structures, as well as input and output mechanisms. Through the assessment of the output created by ChatGPT, we can evaluate its efficacy in facilitating the understanding of foundational Python concepts. Furthermore, we ask the ChatGPT to offer practical coding exercises in Python for novice learners in order to assess its efficacy as an instructor in facilitating programming education through active engagement in practice. Finally, we assess the suitability of ChatGPT for delivering real-time feedback and corrections during the process of debugging a specific issue.

**Prompt:** Provide a roadmap to learn python programming from beginner to basic

|  |  |
| --- | --- |
|  |  |

Figure: Sample Responses

**Prompt:** Provide Hands-On Coding Practice in Python for beginners

|  |  |
| --- | --- |
|  |  |

Figure: Hands on Coding Sample Response

**Prompt:** Real-Time Feedback and Corrections

**Analysis and Discussions**

**Survey Questions**

For the purpose of accurately evaluating the quality of ChatGPT's responses in a wide range of tasks, including interactive programming, providing structured roadmaps, and code debugging, a survey was conducted. The process involved the careful development of a customized collection of survey questions, precisely linked with the main objectives of the proposed framework. The survey questions were thoughtfully developed to closely correspond with the specified research questions, thereby effectively capturing the intended research outcomes. This survey involved a group of three individuals who possess expertise in Python programming. The primary objective of the survey was to use their insightful viewpoints in order to shed light on the quality and efficacy of the responses generated by ChatGPT.

The Likert Scale was employed as the measurement scale for the survey questions. The Likert Scale is widely recognized as a reliable and valid instrument for assessing individuals' subjective opinions and attitudes in an organized manner (Wu & Leung, 2017). The scale encompasses a variety of response alternatives, facilitating participants in expressing their degree of agreement or disagreement with a given proposition. The scale provides respondents with the opportunity to express their degree of agreement or disagreement by assigning a numeric value to their response, which spans from "Strongly Disagree" (1) to "Strongly Agree" (5) (Sullivan & Artino Jr, 2013). The utilization of this scale is common in surveys and questionnaires, facilitating researchers in obtaining a comprehensive understanding of the range of viewpoints and perspectives held by participants. Then, these data may be examined and interpreted to make well-informed decisions.

|  |  |
| --- | --- |
|  |  |

**Figure: Sample Survey Questions**

A brief description of each question is discussed in following section.

1. ChatGPT's Effectiveness in Teaching Python Concepts:

The question aims to assess the efficacy of ChatGPT in instructing specific Python concepts, which is divided into six subparts (I through VI). We inquire with specialists regarding the ability of ChatGPT in instructing new learners on fundamental Python topics such as variables, input and output, and data structures.

1. ChatGPT's Potential for Independent Programming Learning:

This inquiry evaluates the confidence in ChatGPT's potential to facilitate programming education for novices without necessitating human instructors. The question explores the extent to which respondents agree or disagree with the proposition that ChatGPT, an AI-powered tool, may autonomously facilitate the acquisition of programming skills for novices, hence eliminating the necessity for human instructors.

1. Personalizing Learning with ChatGPT:

This inquiry relates to the capacity of ChatGPT to customize the educational experience for novice learners on an individual basis. The participants in the study conveyed their opinions regarding the extent to which ChatGPT is capable of meeting the individual requirements of learners.

1. Timely and Constructive Feedback from ChatGPT:

This question evaluates the ability of ChatGPT to offer constructive and prompt feedback to novices throughout their participation in learning and coding activities.

1. Effectiveness of ChatGPT in Hands-On Coding Practice:

This question seeks to determine the perspectives of experts regarding the effectiveness of ChatGPT in assisting novice individuals in the practical application of coding by creating real code. This inquiry examines the potential of ChatGPT as a dependable companion for learners in facilitating their involvement with practical coding tasks, by seeking respondents' agreement or disagreement.

1. Real-Time Feedback and Corrections from ChatGPT:

This inquiry examines whether ChatGPT has the potential to serve as a valuable mentor by assisting learners in real-time code refinement and debugging. The question aims to gather respondents' perspectives on this matter, seeking agreement or disagreement.

1. Enhancing Proficiency in Comprehension and Application Skills:

This survey question investigates the perspectives of experts regarding the impact of ChatGPT on improving learners' comprehension of programming principles and their practical utilization, with an emphasis on conceptual knowledge and real-world application.

1. Maintaining Bottom-Up Approach in Responses:

This inquiry seeks to determine whether the generated responses of ChatGPT follow the bottom-up methodology, which incorporates starting with fundamental concepts and advancing towards specific details.

1. Maintaining Top-Down Approach in Responses:

In a similar manner, this query assesses whether ChatGPT's responses correspond to a top-down approach, where sample programming examples leads extensive language components.

1. Comparison of Programming Guidance Quality:

The final question seeks to determine if respondents believe that the quality of programming guidance provided by ChatGPT is better compared to the traditional methods, and invites respondents to indicate their level of agreement or disagreement.

**Expert Opinion Analysis**

After obtaining feedback from three Python programming language specialists, we proceed to analyze and compare their respective responses to each survey question. The responses provided by the participants for each survey question were subjected to analysis and comparison in order to identify areas of agreement and disagreement. The combination of expert viewpoints unveils common trends and noteworthy specifics. This procedure ultimately facilitates a comprehensive exploration of expert opinions regarding the efficacy of ChatGPT in the domain of programming education, enabling an analysis of the experts' feedback.

**Figure:**

The Figure illustrates expert responses of the first question of the survey, together with its sub-questions, which assess the ability of ChatGPT in teaching foundational knowledge of Python programming to beginners. In questions labeled as "I," "ii," "v," and "vi," all three experts have provided replies that are generally satisfactory, with ratings ranging from "Agree" (4) to "Strongly Agree" (5). According to the perspectives of these experts, it is evident that ChatGPT possesses the ability to effectively guide inexperienced individuals in learning fundamental Python principles, such as understanding variables and data types, control statements, exception handling, and file handling. In contrast, it is worth noting that two experts expressed a neutral stance, indicating “Neither agree nor disagree” (3), while expert 2 expressed “Agree” (4) in response to question "iii," which pertained to the assessment of ChatGPT's effectiveness in teaching functions and modules. The prevailing viewpoint among experts suggests that there remains potential for enhancing the instructional capabilities of ChatGPT in its provision of teaching functions and modules to novice users. In response to question "iv," all three experts express a neutral stance “Neither agree nor disagree” (3) on ChatGPT's effectiveness in teaching data structures to novices. This suggests that ChatGPT has clear shortcomings in this particular domain. In general, ChatGPT has a satisfactory capability to deliver efficient instruction in teaching the foundational principles of the Python programming language.

**Figure:**

The figure depicts the responses provided by the experts in relation to survey questions "2" to "10". In general, all three experts provide mostly good responses to the majority of the questions. In questions 2, 3, and 4, the expert assigns ratings that span from "Agree" (4) to "Strongly Agree" (5). Based on the provided responses, it is apparent that the respondent express agreement or strong agreement regarding the potential of ChatGPT in facilitating programming education for novices in the absence of instructors. Furthermore, they acknowledge its ability to tailor the learning process to individual beginners and deliver timely and constructive feedback. In response to issue 5, all three experts express a strong agreement with regarding the effectiveness of ChatGPT in facilitating hands-on coding practice for novices. In both question 6 and question 9, all respondents indicated agreement, with an average rate of 4. The observation suggests that ChatGPT possesses the capacity to offer real-time feedback and corrections to individuals who are new to programming. Additionally, it maintains to a top-down approach while generating responses. In relation to questions "8" and "10", it is observed that expert 1 and expert 2 express agreement with a rating of "4", whilst expert 3 expresses a neutral stance with a rating of "3". Based on agreement among experts, it is apparent that ChatGPT follows to a Bottom-Up Approach in its response generation. Furthermore, the programming guidance offered by ChatGPT exhibits superior quality when compared to conventional methodologies. In contrast, the majority of experts express a neutral stance, neither agreeing nor disagreeing (3), regarding question 7. This suggests that ChatGPT has limitations in terms of fostering profound conceptual comprehension and practical application skills among learners, indicating the need for further improvement.

**Conclusion**

**References**

Biswas, S. (2023). Role of ChatGPT in Computer Programming.: ChatGPT in Computer Programming. *Mesopotamian Journal of Computer Science, 2023*, 8-16.

Borji, A. (2023). A categorical archive of chatgpt failures. *arXiv preprint arXiv:.03494*.

Ismail, M. N., Ngah, N. A., & Umar, I. N. (2010). Instructional strategy in the teaching of computer programming: a need assessment analyses. *The Turkish Online Journal of Educational Technology, 9*(2), 125-131.

Isong, B. (2014). A Methodology for Teaching Computer Programming: first year students' perspective. *International journal of modern education and computer science, 6*(9), 15.

Koulouri, T., Lauria, S., & Macredie, R. D. (2014). Teaching introductory programming: A quantitative evaluation of different approaches. *ACM Transactions on Computing Education, 14*(4), 1-28.

Lee, H. (2023). The rise of ChatGPT: Exploring its potential in medical education. *Anatomical Sciences Education*.

Lin, Y., & Weintrop, D. (2021). The landscape of Block-based programming: Characteristics of block-based environments and how they support the transition to text-based programming. *Journal of Computer Languages, 67*, 101075.

Lister, R., Adams, E. S., Fitzgerald, S., Fone, W., Hamer, J., Lindholm, M., . . . Seppälä, O. (2004). A multi-national study of reading and tracing skills in novice programmers. *ACM SIGCSE Bulletin, 36*(4), 119-150.

Lo, C. K. (2023). What is the impact of ChatGPT on education? A rapid review of the literature. *Education Sciences, 13*(4), 410.

Martins, V. F., de Almeida Souza Concilio, I., & de Paiva Guimarães, M. (2018). Problem based learning associated to the development of games for programming teaching. *Computer Applications in Engineering Education, 26*(5), 1577-1589.

McCracken, M., Almstrum, V., Diaz, D., Guzdial, M., Hagan, D., Kolikant, Y. B.-D., . . . Wilusz, T. (2001). A multi-national, multi-institutional study of assessment of programming skills of first-year CS students. In *Working group reports from ITiCSE on Innovation and technology in computer science education* (pp. 125-180).

Medeiros, R. P., Ramalho, G. L., & Falcão, T. P. (2018). A systematic literature review on teaching and learning introductory programming in higher education. *IEEE Transactions on Education, 62*(2), 77-90.

Rahman, M. M., & Watanobe, Y. (2023). ChatGPT for education and research: Opportunities, threats, and strategies. *Applied Sciences, 13*(9), 5783.

Ray, P. P. (2023). ChatGPT: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope. *Internet of Things and Cyber-Physical Systems*.

Saito, D., & Yamaura, T. (2013). *A new approach to programming language education for beginners with top-down learning.* Paper presented at the Proceedings of 2013 IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE).

Saito, D., & Yamaura, T. (2014). *Applying the top-down approach to beginners in programming language education.* Paper presented at the 2014 International Conference on Interactive Collaborative Learning (ICL).

Seng, W. Y., & Yatim, M. H. M. (2014). Computer game as learning and teaching tool for object oriented programming in higher education institution. *Procedia-Social and Behavioral Sciences, 123*, 215-224.

Shidiq, M. (2023). *The use of artificial intelligence-based chat-gpt and its challenges for the world of education; from the viewpoint of the development of creative writing skills.* Paper presented at the Proceeding of International Conference on Education, Society and Humanity.

Subagja, A. D., Ausat, A. M. A., Sari, A. R., Wanof, M. I., & Suherlan, S. (2023). Improving Customer Service Quality in MSMEs through the Use of ChatGPT. *Jurnal Minfo Polgan, 12*(2), 380-386.

Sullivan, G. M., & Artino Jr, A. R. (2013). Analyzing and interpreting data from Likert-type scales. *Journal of graduate medical education, 5*(4), 541-542.

Surameery, N. M. S., & Shakor, M. Y. (2023). Use chat gpt to solve programming bugs. *International Journal of Information Technology & Computer Engineering, 3*(01), 17-22.

Tenenberg, J., & Fincher, S. (2005). Students designing software: a multi-national, multi-institutional study. *Informatics in Education, 4*(1), 143-162.

Tian, H., Lu, W., Li, T. O., Tang, X., Cheung, S.-C., Klein, J., & Bissyandé, T. F. (2023). Is ChatGPT the Ultimate Programming Assistant--How far is it? *arXiv preprint arXiv:.11938*.

Wach, K., Duong, C. D., Ejdys, J., Kazlauskaitė, R., Korzynski, P., Mazurek, G., . . . Ziemba, E. (2023). The dark side of generative artificial intelligence: A critical analysis of controversies and risks of ChatGPT. *Entrepreneurial Business and Economics Review, 11*(2), 7-24.

Whalley, J., & Kasto, N. (2014). *A qualitative think-aloud study of novice programmers' code writing strategies.* Paper presented at the Proceedings of the 2014 conference on Innovation & technology in computer science education.

Wu, H., & Leung, S.-O. (2017). Can Likert scales be treated as interval scales?—A Simulation study. *Journal of social service research, 43*(4), 527-532.

Yilmaz, R., & Yilmaz, F. G. K. (2023). The effect of generative artificial intelligence (AI)-based tool use on students' computational thinking skills, programming self-efficacy and motivation. *Computers and Education: Artificial Intelligence, 4*, 100147.

Yılmaz, R., & Yılmaz, F. G. K. (2022). *Investigation of students’ self-regulation skills, motivation and disorientation in smart mooc*.