Using ChatGPT in Teaching Computer Programming and Studying its Impact on Students Performance

Shubair Abdulla[1], Sameh Ismail[1,2], Yasser Fawzy[1,3] and Abdelrahman Elhaj[1]
[1]Sultan Qaboos University, Oman
[2]Cairo University, Egypt
[3]Helwan University, Egypt

shubair@squ.edu.om
samehsaid@squ.edu.om
yasser.fawzy@squ.edu.om
abdoelhaj@squ.edu.om

https://doi.org/10.34190/ejel.22.6.3380

Abstract: Given the recent emergence of artificial intelligence as an important topic that can contribute to improve curricula and lecture delivery, an increasing number of scholars are investigating its impact in various educational fields. The ChatGPT, which is an artificial intelligence model developed by OpenAI, represents a significant advancement in the generative artificial intelligence area. Since its announcement, integrating the ChatGPT into computer programming curricula - and into other scientific curricula - has yielded some challenges. The main challenge is clearly evident in the conduct of in-class tests and in-lab assignments, where the students are given specific tasks to be accomplished within a certain time frame. As they might seek help from ChatGPT, these types of assessments could be considered a potential threat to academic integrity and may be viewed as a form of academic dishonesty. This study aims at integrating ChatGPT into computer programming curricula, exploring its potential to enhance undergraduate education. It follows a mixed methods approach to examine the potential integration of ChatGPT in teaching computer programming as a supplementary tool. A quasi-experimental design is followed, in which an experimental group is allowed to use ChatGPT and compared to a control group that was not. A research sample of 26 undergraduate students (13 males, 13 females) from the College of Education at Sultan Qaboos University participated in the study. The methodology encompassed three research instruments: in-class exams, in-lab assignments, and semi-structured interviews. These research instruments were utilized to assess the impact of ChatGPT on students' academic performance, which served as both independent (use of ChatGPT) and dependent (student performance) variables. The quantitative analysis revealed a significant enhancement in students' performance, while the qualitative analysis of semi-structured interviews indicated that participants view ChatGPT as a valuable support for learning. Feedback from participants suggested combining ChatGPT with traditional teaching methods to optimize learning outcomes. This study highlights the feasibility and educational benefits of incorporating AI tools like ChatGPT into teaching methodologies. It suggests that such integration can provide a more engaging and effective learning environment, potentially revolutionizing computer programming education. This paper supports e-learning practice by integrating AI-driven tools like ChatGPT into the educational framework and advances the e-learning area by demonstrating these technologies' potential to improve student academic performance in the learning environments. However, the study also acknowledges the need for further research to explore the long-term effects of AI integration in educational settings and to address any emerging challenges. These findings propose a promising direction for future curricular enhancements and suggest an effective method for the integration of AI technologies to support and enrich traditional educational frameworks.

Keywords: ChatGPT, Teaching computer programming, In-class exams, In-lab assignments, Mixed-methods approach

1. Introduction

Computers, smart devices and robots have entered almost all aspects of human life. These devices are currently being utilized in communication, commerce, marketing, teaching and learning, in addition to many other aspects. The importance of programming is evident in building communication interfaces, controlling these digital devices, and managing their systems. The demand for programmers has increased significantly (National Academies of Sciences, Engineering, & Medicine, 2018), and teaching programming has become one of the main goals (Kanika, Chakraverty, & Chakraborty, 2020) of higher education institutions to prepare graduates with programming skills to meet this growing demand. The endeavor of higher education institutions to equip graduates with distinguished programming knowledge and skills has been reflected in the development of curricula and methods for teaching computer programming (Erümit, 2020). As a result, significant efforts have been dedicated to developing all aspects of the educational process. Among the aspects involved in the development are methods for assessing students. In teaching computer programming, assessment plays a crucial role. It enables teachers to evaluate student's abilities to recall facts and basic terms, to measure their

Cite this article: Abdulla, S et al. 2024. "Using ChatGPT in Teaching Computer Programming and Studying its Impact on Students Performance", *Electronic Journal of e-Learning*, 22(6), pp 66-81, https://doi.org/10.34190/ejel.22.6.3380

understanding of codes, and to assess their ability to solve problems using the knowledge and understanding (Erümit, 2020). Moreover, assessing computer programming contributes to developing students' programming skills and enhancing their higher-order thinking (Tikva & Tambouris, 2021).

Artificial Intelligence (AI) has recently become an essential part of education. Many applications of AI have emerged and had a positive impact on teaching, learning, curricula, and assessment (Goksel & Bozkurt, 2019). Among of these is ChatGPT, that is an AI-powered application developed by OpenAI Company. The ChatGPT provides a wide range of services that could be utilized by students such as study materials and in-lab assignments help (Haleem, Javaid & Singh, 2022). However, since its announcement, integrating the ChatGPT into computer programming curricula - and into other scientific curricula - has become a topic of debate (Ali et al., 2023). Some educators consider it a potential threat to academic integrity and others argue that its use in computer programming curricula may be viewed as a form of academic dishonesty (Zielinski et al., 2023). This problem is clearly evident in the conduct of some assessment methods such as in-class tests and in-lab assignments, where the students are given specific tasks to be accomplished within a certain time frame. Examples of tasks as in-class tests and in-lab assignments are analyzing and evaluating existing software, writing programs to solve real-world problems, or researching programming languages and creating reports or presentations (Şenel & Şenel, 2021).

The main goal of this study is to examine the integration of ChatGPT into computer programming curricula. The study attempts to measure the impact of using ChatGPT on students' academic performance and its findings assist educators in making decisions about integrating AI online platforms into curricula. The research questions that this study attempt to address are as follows:

- How does the use of ChatGPT affect students' performance in learning computer programming?
- What are the students' perceptions regarding the use of ChatGPT in learning computer programming?
- What is the effective method for integrating ChatGPT into a computer programming curricula?

To explore the impact of ChatGPT in computer programming education, a quasi-experimental design and semi-structured interviews were incorporated in this study. We compared over 8 weeks the performance of experimental group (allowed to use ChatGPT) with a control group (not allowed to use ChatGPT) utilizing two instruments, in-class exams and in-lab assignments. Participants included 26 second-year undergraduate students of Instructional and Learning Technology major in Sultan Qaboos University (SQU). In order to provide a comprehensive assessment of ChatGPT's effectiveness, a specialized programming course that utilized C# programming language in the MS Visual Studio environment was utilized in this study.

To ensure the integrity of the research and verify that the students in the control group did not use ChatGPT, several measures were implemented. First, students were informed about the research process and the importance of adhering to the research integrity recommendations. We also emphasized that their commitment to not using ChatGPT was essential for producing real and reliable results. Finally, proctoring was conducted during all in-class exams and in-lab assignments to monitor students' activities in the control group.

The remainder of this paper is organized as follows. Section 2 reviews the relevant literature and theoretical frameworks related to the integration of ChatGPT into the computer programming curricula. Section 3 describes the methodology used to collect and analyze data in order to address the research question. Section 4 presents the results of the study along with a discussion of their implications. Finally, Section 5 concludes the study by presenting the key findings, implications, and recommendations for future research.

2. Literature Review

2.1 Using ChatGPT in Teaching Computer Programming

The accuracy of ChatGPT in responding the questions in various fields of science has been widely recognized and led to increase its use in teaching and learning. Its ability to understand and respond to context makes it a valuable learning tool for students (Hassani & Silva, 2023). However, the use of ChatGPT has raised debate about its potential effects on traditional teaching methods. The main issue is that the use of ChatGPT technology may pose a potential threat to academic integrity. Some studies have found that students have begun using ChatGPT and relying more heavily on it to complete essay assignments (Rosenberg, 2023).

Several research studies have examined the use of ChatGPT in diverse fields of education. For instance, Rudolph, Tan, and Tan (2023) discussed the future prospects of ChatGPT in higher education. Their study analyzed its applications for students and teachers, and summarized the potential benefits and risks involved. Similarly,

Baidoo-Anu & Owusu Ansah, L. (2023) recommended the collaborative efforts of policymakers, educators, researchers, and technology experts to make the most of ChatGPT's potential for enhancing education and supporting student learning. Additionally, Yan (2023) explored the implications of ChatGPT on students' writing behavior and learning. His study recommended developing regulatory policies and redefinition of plagiarism to ensure responsible use of ChatGPT.

There are a few papers published to explore the integration of ChatGPT into computer programming curricula. For example, Qureshi (2023) addressed in his paper the use of ChatGPT as a mean of teaching and learning fundamental programming courses in undergraduate Computer Science curricula. The paper discussed the benefits and obstacles of using ChatGPT in this context and examined its impact on students' learning outcomes. The paper findings suggested that students using ChatGPT had an advantage, but there were inconsistencies and inaccuracies in their submitted code, which negatively affected their overall performance and presents challenges that must be proactively managed with an ethical stance. Another paper published by Joshi et al., (2023) aimed to explore the potential impact of ChatGPT on academic integrity in undergraduate computer science education by investigate whether students may be tempted to use ChatGPT to complete take-home assignments and exams, and whether this can negatively affect their learning outcomes. The paper concluded by providing constructive recommendations to both students and instructors on how to use ChatGPT ethically and effectively in academic settings. The authors recommended that instructors educate their students on the potential drawbacks and limitations of ChatGPT and encourage them to use it as a supplementary tool for learning rather than a substitute for acquiring knowledge.

Up to date, the integration of ChatGPT into computer programming curricula has not received much attention in contrast with other educational curricula. Therefore, there is a need for more research to explore the feasibility, effectiveness, and impact of integrating ChatGPT into computer programming curricula, as well as to identify the best practices and strategies for leveraging this technology to enhance students' learning outcomes.

In addition to direct coding assistance, the Generative AI in education provides other tools designed to enrich learning experiences across multiple aspects of computer science education. For instance, Wolfram Alpha (Abramovich, 2021) offers computational intelligence that can serve as a valuable tool for understanding the mathematical foundations of computer science. Another example, Google's BERT, a natural language processing (NLP) mode, can facilitate the development of interactive educational chatbots, offering personalized tutoring and support in learning programming concepts. Moreover, there are several other Generative AI environments that could be utilized for enhancing teaching and learning. These include tools like Codex, developed by OpenAI, which offers capabilities to assist in code generation and debugging. Another notable environment is DeepMind's AlphaCode (Lertbanjongngam et al., 2022) designed to tackle competitive programming challenges and promote problem-solving skills and understanding of algorithms among students. Integrating these Gen AI tools into computer programming curricula can offer diverse perspectives and methodologies for learning and enriching the students' learning experience.

2.2 In-Class Exams and In-Lab Assignments

The in-class exams and in-lab assignments are commonly used assessment tools. They refer to exams and assignments given to students within regular class time. Like scientific curricula, these types of assessment are used in computer programming curricula to evaluate students' understanding of the concepts as well as their ability to apply what they have learned (Bengtsson, 2019). The components of each of these two types of assessments vary; for example, an in-class examination may include written tests and assignments, while an in-lab assignment usually involves a specific task that must be completed within the computer lab. However, these two types are similar in that each is designed to assess a student's mastery of course material (Spiegel & Nivette, 2023).

The use of in-class exams and in-lab assignments along with their impact on the students' academic performance have been addressed in previous studies. For example, Erhel & Jamet (2013) concluded in their research that a regular and timely feedback on in-lab assignments can improve students' understanding of course concepts and increase their motivation to learn. On the other hand, some research shown that the in-class exams can provide a more authentic assessment of students' abilities and allow students to work at their own pace and in a more relaxed environment (Bengtsson, 2019). However, there are some challenges associated with the use of in-class exams and in-lab assignments. Regarding the in-lab assignment, Wang et al. (2012) stated that some students struggle to manage their time effectively, and this often results in last-minute submissions with incorrect answers. The unequal opportunities students to learn is another challenge. For different reasons, some students may not have access to the necessary resources or technology to complete in-class exams and in-lab assignments

(Shakeel et al., 2021). The emergence of ChatGPT has caused another challenge. Some research has pointed to ChatGPT as a potential threat to academic integrity and stated that its use by students for assignments or exams is a form of academic dishonesty (Cotton, Cotton & Shipway, 2023). Despite the challenges associated with their use, the in-class exams and in-lab assignments have a significant impact on students' academic performance. However, further research is needed to explore best practices for using ChatGPT in teaching and learning and to determine the effective ways to support students.

This study targets computer programming students focusing on their performance and perceptions on the integration of ChatGPT into computer programming curricula. Additionally, it attempts to find an appropriate way to integrate ChatGPT as a support learning tool. There are two reasons for this focus. First, ChatGPT has been increasingly used in different fields of education. However, there are still concerns about academic integrity and the appropriate methods to deal with this emerging technology (Joshi et al. 2023). Second, most of the previous studies that examined the integration of ChatGPT into the education field are observational studies and do not involve experiments. Therefore, more research should be conducted to provide more empirical findings (Sallam, 2023).

3. Method

3.1 Research Design

This study utilized a mixed methods approach to investigate the effects of implementing ChatGPT in teaching computer programming on student outcomes. Specifically, a quasi-experimental design was adopted, wherein an experimental group exposed to ChatGPT was juxtaposed against a control group that was not. The study was conducted over a period of 8 weeks, employing three distinct research instruments: in-class examinations, in-lab assignments, and semi-structured interviews. The independent variable in this study was the use of ChatGPT in teaching computer programming. Conversely, the dependent variables encompassed student performance, as evaluated through in-class exams and in-lab tasks.

In light of our experimental design, the following null hypotheses will be tested to find out the impact of ChatGPT on learning outcomes in computer programming education:

- H_0: There is no significant difference in the comprehensive programming concept between the experimental group and the control group.
- H_1: There is no significant difference in the ability to apply programming concepts in hands-on scenarios between the experimental group and the control group.

3.2 Theoretical Framework

This study aims to examine the integration of ChatGPT into computer programming curricula and assess its impact on students' academic performance. To achieve its objectives, this study applies the Unified Theory of Acceptance and Use of Technology (UTAUT) model as a theoretical framework (Venkatesh et al., 2003). The UTAUT framework was developed to examine the use of information systems depending on four key constructs: (1) Performance Expectancy, (2) Effort Expectancy, (3) Social Influence, and (4) Facilitating Conditions. The performance expectancy factor refers to the impact of technology use on improving student's performance. In the context of using ChatGPT in teaching computer programming, this could include the improvement problem-solving abilities, and understanding of programming concepts. The effort expectancy factor refers to level of convenience and ease of technology use. This might involve the ease of interacting with ChatGPT and the user-friendliness of the interface in the context of using ChatGPT in teaching computer programming. The social influence factor measures the levels of importance of technology. In the case of using ChatGPT in teaching computer programming, this might involve the expectations or recommendations from students. Finally, the facilitating conditions factor refers to the availability of an organizational infrastructure to support the use of the technology. With regards of using ChatGPT, this might include the availability of procedures to explain how to use ChatGPT in teaching computer programming.

3.3 Participants

This study was conducted by a quasi-experimental design. A cohort of 26 second-year undergraduate students participated, comprising an equal gender distribution with 13 males and 13 females. All participants were enrolled in the Instructional and Learning Technology major and were undertaking a one-semester course, "Instructional Computer Programming (I)," at the College of Education, Sultan Qaboos University. The research outcomes may be influenced by the academic background of participants. Their familiarity with instructional

technology could make them to more effectively engage with ChatGPT, which might result a more positive reception. The course structure involved two face-to-face classes each week, with each session lasting two hours. These classes were conducted in computer-equipped classrooms. The curricula covered topics related to visual computer programming, including but not limited to: variables, decision statements, iteration statements, classes and objects, arrays, addressing syntax and logic errors, and reading and writing files. Evaluation methods encompassed paper exams, in-class exams, in-lab tasks, and homework assignments. To ensure ethical considerations, students were informed about the study's objectives prior to its commencement, and their informed consent was sought. All students of the course were exposed to equitable learning experiences as all of them consented to participate, which means that our sample is a convenient sample. The sample size of 26 participants represents the total number of enrolled students in the course, which represents the maximum feasible sample for this study. Since, all registered students participated, extending the sample size beyond this number was not possible.

3.4 Procedures

The Instructional Computer Programming (I) course primarily utilized visual programming language within the MS Visual Studio environment. It provides students with a comprehensive platform for learning programming concepts and applying them using C# programming language. This setup ensured a standardized technological context for evaluating the impact of ChatGPT assistance. Data collection covered eight weeks of the SP2023 semester, initiating in the 2nd week and concluding in the 9th week. Quantitative data collection was focused on weeks 2 to 7, while qualitative data were gleaned during weeks 8 and 9. With regards to quantitative data collection, two instruments were employed: in-class exams and in-lab assignments. Each instrument had two distinct versions: for example, in-class exam (1) and in-class exam (2) for the in-class assessment; in-lab assignment (1) and in-lab assignment (2) for the in-lab assignment. The in-class exams assessed students' comprehension of programming concepts, whereas the in-lab assignments tested their capability to apply these concepts in hands-on programming scenarios. Each instrument was deployed twice. The initial deployment split students into two subsets: an experimental group (with access to ChatGPT) and a control group (without ChatGPT access). The same split was followed during the deployment of in-class exam (1). However, in the subsequent in-class exam (2), roles were swapped: students in the original experimental group moved to the control group and vice versa. The same system was applied in the administration of the in-lab assignments, ensuring that each participant experienced both conditions. In addition to offering both groups equitable exposure to ChatGPT, this procedure also minimized potential confounding variables and met ethical guidelines.

Regarding qualitative data, semi-structured interviews were conducted with 16 students selected from the total sample of 26. These students were chosen based on their levels of engagement with ChatGPT and their willingness to provide in-depth feedback, as assessed during in-lab assignments 1 and 2. These interviews aimed to clarify students' perceptions about the efficacy, convenience, and user-friendliness of leveraging ChatGPT in solving in-class exams and in-lab assignments. Although a set of predefined questions guided the interviews, the interviewer allowed flexibility for follow-up questions, elaboration, and exploration of new topics based on the participants' responses.

3.5 Instruments

Three research instruments were used in this study to collect both quantitative and qualitative data:

1. In-class exams: These exams were administered twice: first in the 4th week and later in the 7th week. The initial exam aimed to assess participants' comprehension of the concepts related to graphical user interface, event handling, and variables and constants, while the second focused on functions and classes, and sub-procedures. Both tests presented essay-style questions, demanding written answers to demonstrate their understanding of the topic. The questions were developed based on the learning objectives and content of Instructional Computer Programming (I) course. Figures 1 and 2 show the in-class exams used as instruments research in this study.
2. In-lab assignments: These tasks were assigned on two occasions: in the 3rd week and the 6th week. The first assignment tasked participants with a decision-making exercise, specifically focusing on the 'If-blocks' concept. The subsequent assignment directed participants to solve problems related to arrays and file handling. Participants were instructed to design and develop complete programs and submit their code solutions for evaluation. The tasks were designed to align with the learning objectives and content. Figures 3 and 4 present the in-lab assignment used in this research.
3. Semi-structured interviews: Semi-structured interviews were conducted with a select group of participants to collect their perceptions of using ChatGPT, specifically its convenience and efficacy in

approaching in-class exams and in-lab assignments. The interview framework was established based on UTAUT framework and a thorough literature review. A set of 5 questions (shown in Table 1) was used to guide the interviews. Prior to the main study, a pilot test was conducted with a smaller participant group of 4 students to fine-tune the interview questions.

TECH3222 - Instructional Computer Programming I

In-class exams (1)

An application has been created to convert one kilometer to inches, feet, yards, and miles as in the screenshot below. When a button is clicked, the label has an equivalent number to one kilo, and when the 'Exit' button is clicked, the application terminates. You are required to write the click event for all five buttons.

Tip:

1 Kilometer = 3,281 feet, 1 Kilometer = 39,370 inches, 1 Kilometer = 0.6214 miles, and 1 Kilometer = 1,093.6 yards.

Assessment Rubric

#	Criteria	Unsatisfactory (0)	Needs Improvement (1)	Good (2)
1	Inches button click event			
2	Feet button click event			
3	Yards button click event			
4	Miles button click event			
5	Exit button click event			
			Total Earned Points:	

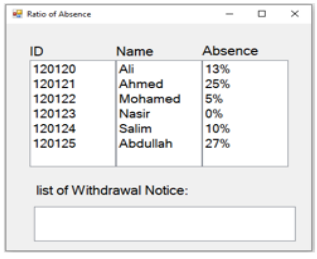
Figure 1: The first in-class exam - variables and constants concepts

TECH3222 - Instructional Computer Programming I

In-class exams (2)

Design a program that checks the ratio of absence for all students and shows on the form a list "Absentee Withdrawal Notice" for all the students with ratio more than 20%. The design and input are shown below

Ratio of Absence for the Students



ID:	120120	120121	120122	120123	120124	120125
Name:	Ali	Ahmed	Mohamed	Nasir	Salim	Abdullah
absence:	13%	25%	5%	0%	10%	27%

Assessment Rubric

#	Criteria	Unsatisfactory (0)	Needs Improvement (1)	Good (2)
1	Input and output			
2	Sub procedures			
3	functions			
4	Loop			
5	Conditions			
			Total Earned Points:	

Figure 2: The second in-class exam - functions and sub procedures concepts

TECH3222 - Instructional Computer Programming I

In-Lab Assignments (1)

Telling which part of the day it is

You are given the task of designing a program that tells you which part of the day it is, given a user-entered time (hour & minute). Please use the 24h clock. Show the user two input text boxes, one for the hour of day, the other for the minute. The user then clicks a "Convert" button, your program should display the corresponding time of day in the color specified below. Use the following table:

Hour	Time of Day	Color
21:00-23:59 or 0:00-5:59	Night	Black
6:00-10:59	Morning	Yellow
11:00-13:59	Lunch	Orange
14:00-17:59	Afternoon	Light Blue
18:00-20:59	Evening	Dark Blue

Assessment Rubric

#	Criteria	Unsatisfactory 0	Needs Improvement 1	Good 2
1	Graphical User Input (GUI) Design			
2	Using Datatype Conversion			
3	Using If Statements and select case block			
4	Validation of User Inputs			
5	The comparison between if then elseif block and select case block			
			Total Earned Points:	

Figure 3: The first in-lab assignment - decision (If-blocks) concepts

TECH3222 - Instructional Computer Programming I

In-Lab Assignments (2)

Students' Grade book

Develop a VB.NET program that reads from a CSV file (st_performance.csv) a list of student marks. The list contains 28 students of programming course section 10, and stores the midterm mark (20%), assignment 1, 2, 3, and 4 marks (10% each), and final exam mark (40%). An example is shown in the image below.

```
Ahmed,16.5,10,8,6,10,33.5
Amjad,19.5,10,10,10,9,36
Basim,15,7.5,7.5,8,9,37
Hashim,12,8,7,6,9,30
Qusay,15.5,10,10,10,9,36.25
```

The program then defines sub programs or functions for:

1. Calculates and displays the total score for each student
2. Calculates and displays the grade for each student (A, A-, B+, B, …)
3. Calculate and display the percentage of success and failure in the section
4. Calculate and display the percentage of success and failure in the final exam
5. Calculate and display the percentage of grades A, A-, B+, B, …
6. Calculate and display the maximum mark
7. Calculate and display the minimum mark

The program must provide an option to save the calculation results to a text file. The image below shows an example of saving the results of calculations.

Assessment Rubric

#	Criteria	Unsatisfactory 0	Needs Improvement 1 or 0.5	Good 2 or 1
1	Calculates the total score for each student **(1 marks)**			
2	Calculates the grade for each student (A, A-, B+, B, …) **(2 marks)**			
3	Calculate the percentage of success and failure in the section **(2 marks)**			
4	Calculate the percentage of success and failure in the final exam **(2 marks)**			
5	Calculate the percentage of grades A, A-, B+, B, … **(1 marks)**			
6	Calculate the maximum mark **(1 marks)**			
7	Calculate the minimum mark **(1 marks)**			
			Total Earned Points:	

Figure 4: The second in-lab assignment - arrays and files concepts

Table 1: Question of the Semi-structured interview

#	Question
1	What types of tasks and challenges do you typically try to overcome when utilizing ChatGPT for your learning in computer programming?
2	In what ways has ChatGPT been helpful in your learning journey of computer programming?
3	Have you encountered any limitations or drawbacks when using ChatGPT for learning programming?
4	How do you think ChatGPT compares to more traditional methods of learning programming, such as textbooks, online tutorials, or instructor-led classes?
5	Do you believe ChatGPT should be integrated into computer programming education?

3.6 Validity and Reliability

To ensure a high degree of research dependability, several measures were implemented to reinforce validity and reliability. For the in-class exams and in-lab assignments, content validity was assessed. All versions of these instruments were initially designed by the course instructor. The instruments were reviewed by two course coordinators to confirm the appropriateness of their content. An additional measure of reliability, the test-retest method, was applied to the in-class exams and in-lab assignments. The goal of this method was to assess the consistency of these instruments. Since these instruments are parts of student grades, the test-retest was conducted on 10 students from different cohorts that are separate from the main study group. A two-week interval was set between the initial test and its subsequent retest. After recording the scores from both tests, a correlation coefficient was calculated, yielding a value of 0.73. This result suggests that the in-class exams and in-lab assignments exhibit an acceptable level of reliability. As for the semi-structured interview instrument, two specialists in the domain reviewed the instrument, offering their perspectives and recommendations. Based on their feedback, some modifications were made to the interview questions, which included amending three questions and deleting one question.

3.7 Data Analysis

The quantitative data analysis in this study involved the examination of student's performance from the in-class exam and the in-lab-assignment instruments. The purpose of data analysis was to assess the impact of using ChatGPT on students' performance in computer programming. Specifically, the data analysis process focused on comparing the performance of two groups: the experimental group (using ChatGPT) and the control group (not using ChatGPT). The raw data collected from the in-class exams and in-lab assignments was gathered and organized for analysis. This data consists of the scores obtained by students. The students' scores in these two exams are set at 100 to align with the evaluation system used in the Instructional Computer Programming (I) course. The data was thoroughly examined to ensure completeness, accuracy, and consistency.

Descriptive statistics were calculated for the in-class exam scores and in-lab assignment scores separately for both the experimental and control groups. Measures such as mean, standard deviation, and range were computed for each group to provide an overview of their performance. A comparative analysis was conducted to assess the differences in performance between the experimental and control groups. Initially, the Shapiro-Wilk test was performed for each group to assess normality. Based on the results of the Shapiro-Wilk tests, statistical tests such as independent samples t-tests or Mann-Whitney U tests (Royston, 1992; Sawilowsky, 2005) were conducted to determine if there were significant differences in the exam scores and assignment scores between the two groups.

Descriptive statistics were generated using MS Excel, while the Python programming language facilitated the comparative analysis. Specifically, within Python, the Shapiro-Wilk test, independent samples t-test, Mann-Whitney U test, and calculation of Cohen's d effect size measure were conducted. The qualitative data were collected through semi-structured interviews conducted with a subset of participants. The interviews aimed to explore the students' perceptions regarding the convenience and ease of using ChatGPT for solving in-class exams and in-lab assignments. The data analysis process for the qualitative data involved the following steps:

- The transcriptions were reviewed to develop a deeper understanding of the participants' perspectives.
- A thematic coding approach (Shoufan, A. 2023) was employed to identify patterns, themes, and concepts within the data.

- A coding scheme was developed based on the identified themes and patterns. The coding scheme provided a structured framework for organizing and categorizing the data.
- The coded data were summarized, condensed, and organized to capture the essential information related to the research questions. This process involved the extraction of relevant quotes, illustrative examples, or excerpts that represented the identified themes and supported the interpretation of the findings.

The qualitative data analysis process was facilitated by the use of MS Word, a word processor to aid in organizing, coding, and managing the data. The program allowed the efficient arrangement of the text and encrypted passages in tables in order to carry out the comprehensive analysis process.

4. Results and Discussion

The primary objective of this research is to examine the integration of ChatGPT into computer programming curricula and assess its impact on students' academic performance. Guiding this investigation are the following research questions:

- How does the use of ChatGPT affect students' performance in learning computer programming?
- What are the students' perceptions regarding the use of ChatGPT in learning computer programming?
- What is the effective method for integrating ChatGPT into a computer programming curricula?

The following null hypotheses were also formulated:

- H_0: There is no significant difference in the comprehensive programming concept between the experimental group and the control group.
- H_1: There is no significant difference in the ability to apply programming concepts in hands-on scenarios between the experimental group and the control group.

The subsequent sections will present and discuss findings pertinent to these questions.

4.1 Effect of ChatGPT on the Performance of Computer Programming Leaners

4.1.1 Descriptive statistical analysis

To provide a snapshot of the overall performance, Table 2 presents the mean, standard deviation, minimum, and maximum scores for both groups across the two tests, the in-class exam and the in-lab-assignment. In the first in-class exam, the experimental group averaged 78.85 compared to the control's 58.08. The smallest difference between means is seen in the second in-class exam, with scores of 62.31 for the experimental group and 63.85 for the control group. Additionally, the range of scores indicates a generally wider variation in the control group, especially in the second in-class exam, where the maximum score reached 90 compared to the experimental group's 80. Table 3 shows details about the individual scores for application 1 and application 2 for the In-Class Exam and In-Lab Assignment. Scores within the control (C) and experimental (E) groups varied. For Application#1 exams, control group scores mostly ranged between 40 and 70, while the experimental group often surpassed 90. The experimental group generally scored higher than the control in in-lab assignments, particularly for Application#2.

Table 2: Descriptive statistics of scores for both groups across in-class exams and in-lab assignments

Group	Instrument/application	Mean	STD	Range	
				Min	Max
Experimental Group	In-Class Exam (1)	78.85	12.75	57	99
	In-Class Exam (2)	62.31	11.20	40	80
Control Group	In-Class Exam (1)	58.08	12.02	40	80
	In-Class Exam (2)	63.85	15.95	40	90
Experimental Group	In-Lab Assignment (1)	80.77	9.17	70	100
	In-Lab Assignment (2)	84.62	6.34	70	90
Control Group	In-Lab Assignment (1)	69.23	7.30	60	80
	In-Lab Assignment (2)	76.92	7.22	70	90

4.1.2 Inferential statistics analysis

While descriptive statistics offer a preliminary insight into the performance of both the experimental and control groups, they do not provide a comprehensive understanding of the underlying differences between the groups. Therefore, inferential statistical analysis is crucial. This section will detail the results of the inferential statistical tests that were conducted to more robustly assess the impact of integrating ChatGPT into computer programming learning. Assessing the normality of score distributions for both groups is a foundational step in inferential statistical analysis. The Shapiro-Wilk test, commonly use to check for normal distribution, suggests the that the data follows a normal distribution. A p-value below a predetermined alpha level (usually set at 0.05) indicates that the data is not normally distributed. Table 4 presents the results of this Shapiro-Wilk test for both the experimental and control groups.

Table 3: Individual scores for application 1 and application 2 in in-class exams and in-lab assignments

Application#1				**Application#2**			
Student #	**Group**	**In-Class Exam (1)**	**In-Lab Assignment (1)**	**Student #**	**Group**	**In-Class Exam (2)**	**In-Lab Assignment (2)**
1	C	70	60	26	C	70	80
2	C	65	70	25	C	60	80
3	C	50	80	24	C	50	70
4	C	55	60	23	C	80	70
5	C	80	70	22	C	70	90
6	C	70	70	21	C	40	80
7	C	60	70	20	C	90	70
8	C	65	80	19	C	60	70
9	C	50	60	18	C	50	70
10	C	45	70	17	C	40	80
11	C	40	60	16	C	60	80
12	C	65	70	15	C	70	90
13	C	40	80	14	C	90	70
14	E	90	80	13	E	70	80
15	E	58	100	12	E	40	90
16	E	77	80	11	E	60	80
17	E	60	70	10	E	50	90
18	E	99	80	9	E	50	90
19	E	94	80	8	E	70	90
20	E	57	80	7	E	60	90
21	E	80	90	6	E	80	80
22	E	85	70	5	E	60	80
23	E	85	70	4	E	60	90
24	E	80	90	3	E	60	90
25	E	80	70	2	E	70	80
26	E	80	90	1	E	80	70

Table 4: Shapiro-Wilk test results

Tool/Application	P-Values	
	Experimental	Control
In-Class Exam (1)	0.140	0.570
In-Class Exam (2)	0.390	0.430
In-Lab Assignment (1)	0.070	0.010
In-Lab Assignment (2)	0.002	0.005

For the In-Class Exam (1 & 2) and the In-Lab Assignment (1), p-values are above the typical threshold (e.g., 0.05), indicating that the data sets are normally distributed. On the other hand, the p-values for the In-Lab Assignment (2) are below this threshold for both groups, which means that this particular data set is not normally distributed.

Based on the results of the Shapiro-Wilk normality test and in accordance with standard statistical practices, the Independent Samples t-test was chosen for analyzing the scores of In-Class Exam (1 & 2) and In-Lab Assignment (1). Given that the data for In-Lab Assignment (2) did not exhibit normal distribution, the Mann-Whitney U test, a non-parametric alternative to the Independent Samples t-test, was employed for this specific dataset. The results of the inferential statistics are presented in the following:

In-Class Exam (1):

- Independent Samples t-test:
- *t-value: -4.1074*
- *Degrees of Freedom (df): 24*
- *p-value: 0.0004*
- Interpretation: The resulting p-value of 0.0004 is much less than the typical significance threshold of 0.05. The results indicate a statistically significant difference in the scores between the experimental and control groups. The negative t-value suggests that the mean of the experimental group (78.85) is greater than the mean of the control group (58.08) as shown in Table 1. This difference is statistically significant ($p < 0.05$), meaning the experimental group performed significantly better than the control group in the In-Class Exam (1).

In-Class Exam (2):

- Independent Samples t-test:
- *t-value: 0.2734*
- *Degrees of Freedom (df): 24*
- *p-value: 0.7869*
- Interpretation: The resulting p-value of 0.7869 is higher than the typical significance threshold of 0.05. This means there's no statistically significant difference in the scores of the experimental group compared to the control group for this exam. The performance of both groups is statistically similar for In-Class Exam (2).

In-Lab Assignment (1):

- Independent Samples t-test:
- *t-value: -3.4114*
- *Degrees of Freedom (df): 24*
- *p-value: 0.0023*
- Interpretation: The resulting p-value is 0.0023, which is below the typical significance threshold of 0.05. This result indicates a statistically significant difference between the scores of the two groups. The negative t-value indicates that the mean score of the experimental group (80.77) is greater than the mean score of the control group (69.23) as shown in Table 1. Given that the p-value is less than 0.05, this difference is statistically significant. Hence, the experimental group outperformed the control group in In-Lab Assignment (1).

In-Lab Assignment (2):

- Mann-Whitney U test:
- *U-value: 39.5*

- *p-value: 0.0152*
- Interpretation: The U-value here indicates a difference in the ranking of scores between the two groups. Since the mean of the experimental group (84.62) is greater than the mean of the control group (76.92) for the In-Lab Assignment (2) as shown in Table 1 and the p-value is less than 0.05, we can conclude that the experimental group performed significantly better than the control group in In-Lab Assignment (2).

The null hypotheses (H_0 and H_1) suggested that there are no significant differences in performance between the experimental and control groups in terms of in-class exam and in-lab assignment outcomes. The statistical analyses conducted, the Sample t-tests and Mann-Whitney U test led to the rejection of the null hypothesis for the first in-class exam (H_0) and the in-lab assignments (H_1). On the other hand, the null hypothesis was not rejected for the second in-class exam, which suggests that no significant difference in performance between the two groups in that context only.

4.1.3 Discussion

As the results of the descriptive and inferential statistics analyses suggest, the use of the ChatGPT appears to have a positive impact on the performance of computer programming students in the contexts studied. In the In-Class Exam (1), students using ChatGPT significantly outperformed those who did not. This is likely because the content of this exam was more general in nature, and such broad topics (converting kilometers to feet or inches) are areas that ChatGPT is undoubtedly trained on. While the control group primarily depended on their memory and understanding of the concepts, the experimental group depended on ChatGPT for answers which could easily address such general questions. On the other hand, In-Class Exam (2) presented a unique scenario. Despite its focus on functions and sub-procedures, the questions were customized to a specific topic that is: addressing absenteeism at Sultan Qaboos University. Such specialized contexts might not be readily available in ChatGPT's training data, and it would require specific prompting or expertise to get the most out of the ChatGPT. Hence, there was no statistically significant difference between the two groups in this exam. This shows that while ChatGPT can be a valuable learning tool for general programming queries, its effectiveness may vary when faced with highly specific or localized questions. This implies that in certain contexts or situations, the usage of ChatGPT might not necessarily provide a distinct advantage.

In regard to the practical applications observed in In-Lab Assignment (1) and In-Lab Assignment (2), the data indicates that students who used ChatGPT had a clear performance advantage over those who did not. During In-Lab Assignment (1), students were asked to solve a general problem commonly faced by many: determining which part of the day it is using decision-based (If-blocks) concepts. Given the broad and common nature of this assignment, it's reasonable that ChatGPT had been trained on similar queries. The results in In-Lab Assignment (1) is similar to the results observed in In-Class Exam (1). However, In-Lab Assignment (2) introduced a more specific challenge, where students were asked to compute the grade book of students at Sultan Qaboos University. Though the experimental group did outperform the control group, the difference in means (84.62 for the experimental group vs. 76.92 for the control group) is relatively small. This narrow performance gap can be potentially attributed to the fact that students had access to Visual Studio IDE, enabling them to write, debug, and optimize their programs before submission. Such resources may have lessened the distinct advantage offered by ChatGPT in this particular scenario.

4.1.4 Response to the research question: How does the use of ChatGPT affect students' performance in learning computer programming?

In response to this question, based on the results of the descriptive analysis, we can conclude that the influence of the ChatGPT on computer programming students' performance is context dependent. For examinations, if questions entail solving highly customized problems, ChatGPT may not offer a noticeable advantage. In practical tests, while ChatGPT does contribute to enhanced performance, the advantage is slightly reduced since students utilize editors and compilers like Visual Studio IDE to refine their programs before submission. While ChatGPT can be beneficial for general programming queries, its impact decreases in situations demanding specialized solutions or when other optimization tools are accessible to students.

4.2 Students' Perceptions Regarding the Use of ChatGPT in Learning Computer Programming

4.2.1 Qualitative analysis

For the qualitative analysis, which aimed at understanding students' perceptions regarding the use of ChatGPT in learning computer programming, semi-structured interviews were employed as the primary data collection method. This format granted flexibility, enabling participants to share detailed opinions and perceptions. At the same time, adherence to a set of predefined questions ensured consistency across all interviews. From the total sample of 26 students, 16 were chosen for these interviews. The 16 students were selected based on their levels of engagement with ChatGPT and their willingness to provide in-depth feedback, as assessed during the conduct of in-lab assignments 1 and 2. The remaining 10 students were not included in the interviews due to their limited availability for participation, which is caused by the lower levels of engagement with ChatGPT or scheduling conflicts. This selection ensured a diverse range of experiences and perspectives were represented, especially concerning the utilization of ChatGPT to complete assignments, homework, and take-home exams. Table 5 shows the examples of codes, theme, along with some students quotes collected from the interviews.

Table 5 Themes, codes, and examples of student responses

Theme	Codes	Examples from Student Responses
Impact of ChatGPT on Student Performance	Performance improvement, exam outcomes, lab assignments, performance variation	"Using ChatGPT helped me quickly understand the problem during the exam, which improved my performance."
Context Dependency of ChatGPT's Effectiveness	General vs. specialized queries, resource availability, exam type	"For general questions, ChatGPT was really helpful, but it didn't provide advantage for the specialized questions in our second exam."
Limitations and Drawbacks of ChatGPT	Limitations, student challenges, information management	"ChatGPT provides huge information about programming when I'm prompting it. It's not easy to use it in certain situations."
Comparison of ChatGPT to Traditional Learning Methods	Traditional vs. AI-based learning, integration with existing tools	"We need textbooks, online tutorials, and instructor-led classes to be prepared for midterm exams and final exams."
Integration of ChatGPT into Educational Curricula	Curriculum design, educational technology, AI integration strategies	"We need to use ChatGPT in learning programming as it gives immediate and direct response, but we need to monitor its use to prevent cheating."

4.2.2 Students' experiences and perceptions of ChatGPT in programming education

To understand the role of ChatGPT in computer programming education, several key themes and patterns emerged from the semi-structured interviews. The qualitative analysis highlights a spectrum of student experiences, ranging from the tasks and challenges they faced, the help they got from ChatGPT, the limitations of ChatGPT, and how ChatGPT compares to traditional educational methods.

Impact of ChatGPT on Student Performance:

For many students interviewed, the served as a supportive tool, especially when solving assignments, writing reports, and searching for appropriate data structures to save the data used in developed programs. One student shared, “I am using ChatGPT frequently because it gives me the proper data structures like dictionary, list, and tuple to be used in saving my data.” This reflects the platform's capacity to provide immediate and actionable insights that can streamline the coding process.

Context Dependency of ChatGPT's Effectiveness:

In regard to the tasks and challenges, students found ChatGPT useful in guiding them through the programming libraries. One student noted, “I usually use the ChatGPT to know which library I should use to complete the program because the programming language has many libraries.” This indicates a reliance on ChatGPT for navigating the language libraries, which can often be overwhelming for new students. Some other common challenges in the responses are explaining code, searching for the appropriate syntax, identifying the right libraries to import, and pinpointing runtime errors in their programs.

Limitations and Drawbacks of ChatGPT:

However, the tool wasn't without its limitations. A student remarked, “ChatGPT provides huge information about the programming when I'm prompting it. It's not easy to use it in certain situations.” This highlights a potential issue where the information overload can make it challenging to quickly extract specific solutions.

Comparison of ChatGPT to Traditional Learning Methods:

The interviews also provided comparison of ChatGPT with traditional learning tools. While students saw value in ChatGPT for specific tasks, they emphasized the essential role of textbooks, online tutorials, and instructor-led classes. “We need textbooks, online tutorials, and instructor-led classes to prepare ourselves for midterm exams and final exams,” commented one student. Another student added, “The ChatGPT is useful in preparing ourselves for exams because we use it to solve examples provided by the instructor in the PPTs and textbooks.” These quotes reflect that while ChatGPT is a valuable asset, it serves as a complement rather than a replacement for traditional learning tools.

Integration of ChatGPT into Educational Curricula:

Regarding the potential integration of ChatGPT into educational curricula, there was a clear feeling towards its inclusion. “We need to use the ChatGPT in learning programming as it gives immediate and direct response,” shared a student. However, some expressed difficulties, emphasizing the importance of oversight: “We need these emerging technologies like AI in teaching and learning, but we need to monitor the use of it to prevent cheating.” This highlights the balancing required to ensure that while students benefit from advanced tools, the integrity of the learning process should be considered.

4.2.3 Response to the research question: What are the students' perceptions regarding the use of ChatGPT in learning computer programming?

Based on the qualitative findings from the semi-structured interviews, students' perceptions regarding the use of ChatGPT in learning computer programming are generally positive. They considered ChatGPT as a valuable asset in learning computer programming. It offers real-time guidance and support. However, they also emphasized the importance of its role as a supplementary tool, working alongside traditional teaching methods, rather than replacing them. The primary sentiment was that while ChatGPT can enhance the learning experience, it should be used wisely, with the appropriate checks in place to ensure academic integrity.

4.3 The Effective Method for Integrating ChatGPT into a Computer Programming Curricula

Based on both quantitative and qualitative data analyses, the effective approach for integrating ChatGPT into a computer programming curricula in determined in this section. The mixed methodologies followed in this study helped provide a comprehensive understanding of how ChatGPT can be integrated without compromising the core essence of traditional teaching. The detailed clarification of this integration strategy is presented below.

4.3.1 Response to the research question: What is the effective method for integrating ChatGPT into a computer programming curricula?

The answer to these questions, stemming from both quantitative and qualitative data analyses, requires a blend of state-of-the-art tools and the traditional pedagogies. ChatGPT, despite its impressive capabilities, should not replace the foundational instructional methods. Instead, it should serve as an additive component, amplifying the learning environment. The challenge in integrating ChatGPT lies in finding the right balance: granting students access to ChatGPT while keeping the integrity. This challenge becomes evident in scenarios such as exams and assignments, where there's a possibility for unsupervised use of ChatGPT, moving students away from the traditional oversight of human proctors or advanced monitoring systems. The solution to this challenge is designing these assessments using localized or customized problems. Such an approach ensures that learning maintains its integrity, even as students utilize the power of contemporary tools like ChatGPT.

5. Conclusion

This study examined the impact of ChatGPT on computer programming students' performance in in-class exams and in-lab assignments. It also explored the computer programming students' perceptions regarding the interaction of ChatGPT into the computer programming curricula. Furthermore, the research aimed to identify the effective manner to integrate ChatGPT into computer programming curricula as a supplementary learning tool. In examining the implications of integrating ChatGPT into computer programming instruction, the study

employed a mixed-methods, qualitative and quantitative, strategy. A quasi-experimental design was applied, contrasting an experimental group using ChatGPT with a control group that did not.

The results of the quantitative analysis showed that the integration of ChatGPT into the learning process has a pronounced impact on students' performance. The experimental group, which utilized ChatGPT, consistently displayed superior performance in both in-class exams and in-lab assignments when compared to the control group. This underscores the potential of ChatGPT as a valuable supportive learning tool in computer programming education. The superior scores of the experimental group suggest that ChatGPT may have facilitated a deeper understanding of computer programming concepts. The platform's ability to provide immediate feedback, clarify doubts, and present information in a user-friendly manner might have contributed to this enhanced understanding (Grassini, 2023).

Qualitative feedback from the semi-structured interviews revealed that students predominantly viewed the integration of ChatGPT in computer programming education positively. They appreciated the real-time guidance it provided, underscoring its merit as a supplementary tool rather than a substitute for conventional teaching methods (Shoufan, 2023).

An effective method to integrate ChatGPT into a computer programming curricula could be concluded from quantitative and qualitative analyses. Initially, it is important to set a balance between using modern tools and traditional teaching methods. While ChatGPT can amplify the learning environment, it should not replace foundational instructional techniques. The primary challenge is in situations like exams and assignments, where there is potential for unsupervised usage of ChatGPT. To address this challenge, assessments should be designed with localized or customized problems. This ensures that the learning process retains its integrity, even when students leverage advanced tools like ChatGPT.

The noticeable improvement in the performance of students who used ChatGPT suggests that the integration of such AI-driven platforms can lead to more effective learning outcomes. Institutions and educators should consider incorporating ChatGPT and similar tools to capitalize on their potential to facilitate a deeper understanding of complex subjects like computer programming. The positive feedback from students regarding ChatGPT's integration underscores its merit, but it also highlights the need for a balanced educational approach. It's crucial that while AI tools are integrated, foundational teaching methods are preserved. Furthermore, to maintain the integrity of assessments, educators should prioritize crafting localized or customized problems especially when the potential for unsupervised tool usage exists.

While the current study provides insights into the benefits of ChatGPT, there's potential for further research. This could explore the long-term impacts of using ChatGPT, its effectiveness at different stages in the study program, and its utility in other domains beyond computer programming.

References

Ali, J. K. M., Shamsan, M. A. A., Hezam, T. A. & Mohammed, A. A. (2023). Impact of ChatGPT on Learning Motivation: Teachers and Students' Voices. Journal of English Studies in Arabia Felix, 2(1), 41-49. https://doi.org/10.56540/jesaf.v2i1.51

Abramovich, S. (2021). Using Wolfram Alpha with elementary teacher candidates: From more than one correct answer to more than one correct solution. Mathematics, 9(17), 2112. https://doi.org/10.1080/0020739X.2010.493241

Baidoo-anu, D. & Owusu Ansah, L. (2023). Education in the Era of Generative Artificial Intelligence (AI): Understanding the Potential Benefits of ChatGPT in Promoting Teaching and Learning. Journal of AI, 7 (1), 52-62. Retrieved from https://dergipark.org.tr/en/pub/jai/issue/77844/1337500 https://doi.org/10.61969/jai.1337500

Bengtsson, L. (2019). Take-home exams in higher education: A systematic review. Education Sciences, 9(4), 267.

Cotton, D. R. E., Cotton, P. A., & Shipway, J. R. (2023). Chatting and cheating: Ensuring academic integrity in the era of ChatGPT. Innovations in Education and Teaching International, 1–12. https://doi.org/10.1080/14703297.2023.2190148

Erümit, Ali Kürşat. "Effects of different teaching approaches on programming skills." Education and Information Technologies 25, no. 2 (2020): 1013-1037. https://doi.org/10.1007/s10639-019-10010-8

Erhel, S. and Jamet, E. (2013) 'Digital game-based learning: Impact of instructions and feedback on motivation and learning effectiveness'. Computers & education, 67, pp. 156–167. https://doi.org/10.1016/j.compedu.2013.02.019

Grassini, S. (2023). Shaping the future of education: exploring the potential and consequences of AI and ChatGPT in educational settings. Education Sciences, 13(7), 692. https://doi.org/10.3390/educsci13070692

Goksel, N., & Bozkurt, A. (2019). "Artificial intelligence in education: Current insights and future perspectives". In Handbook of Research on Learning in the Age of Transhumanism, pp. 224–236. IGI Global. https://doi.org/10.4018/978-1-5225-8431-5.ch014

Haleem, A., Javaid, M., & Singh, R. P. (2022). An era of ChatGPT as a significant futuristic support tool: A study on features, abilities, and challenges. BenchCouncil Transactions on Benchmarks. Standards and Evaluations, 2(4), 100089. https://doi.org/10.1016/j.tbench.2023.100089

Hassani, H. and Silva, E. S. (2023). The role of ChatGPT in data science: how AI-assisted conversational interfaces are revolutionizing the field. Big data and cognitive computing, 7(2), 62-78. https://doi.org/10.3390/bdcc7020062

Joshi, I., Budhiraja, R., Dev, H., Kadia, J., Ataullah, M. O., Mitra, S., ... & Akolekar, H. D. (2023). ChatGPT--A blessing or a curse for undergraduate computer science students and instructors. arXiv preprint arXiv:2304.14993. https://doi.org/10.48550/arXiv.2304.14993

Kanika, Chakraverty, S., & Chakraborty, P. (2020). Tools and techniques for teaching computer programming: A review. Journal of Educational Technology Systems, 49(2), 170-198. https://doi.org/10.1177/0047239520926971

Lertbanjongngam, S., Chinthanet, B., Ishio, T., Kula, R. G., Leelaprute, P., Manaskasemsak, B., ... & Matsumoto, K. (2022, October). An empirical evaluation of competitive programming ai: A case study of alphacode. In 2022 IEEE 16th International Workshop on Software Clones (IWSC) (pp. 10-15). IEEE. https://doi.org/10.1109/IWSC55060.2022.00010

National Academies of Sciences, Engineering, and Medicine (2018). Appendix C: Commissioned Paper: Workforce Trends in Computer Science. In Assessing and Responding to the Growth of Computer Science Undergraduate Enrollments. Washington, DC: The National Academies Press. https://doi.org/10.17226/24926

Qureshi, B. (2023). Exploring the use of ChatGPT as a tool for learning and assessment in undergraduate computer science curriculum: Opportunities and challenges. arXiv preprint arXiv:2304.11214. https://doi.org/10.48550/arXiv.2304.11214

Rosenberg, M. (2023, January 16). AI Writing Programs Raise Questions About Cheating in Universities. The New York Times. https://www.nytimes.com/2023/01/16/technology/chatgpt-artificial-intelligence-universities.html

Royston, P. (1992). Approximating the Shapiro-Wilk W-test for non-normality. Statistics and computing, 2, pp. 117–119. https://doi.org/10.1007/BF01891203

Rudolph, J., Tan, S., & Tan, S. (2023). ChatGPT: Bullshit spewer or the end of traditional assessments in higher education? Journal of Applied Learning and Teaching, 6(1), 1-22. https://doi.org/10.37074/jalt.2023.6.1.9

Sallam, M. (2023). ChatGPT utility in healthcare education, research, and practice: Systematic review on the promising perspectives and valid concerns, Healthcare, 11(6), 887. https://doi.org/10.3390/healthcare11060887

Sawilowsky, S. S. (2005) 'Misconceptions leading to choosing the t test over the Wilcoxon Mann-Whitney test for shift in location parameter'. Journal of Modern Applied Statistical Methods, 4(2), 26. https://doi.org/10.22237/jmasm/1130804700

Shakeel, A., Shazli, T., Salman, M. S., Naqvi, H. R., Ahmad, N., & Ali, N. (2021). Challenges of unrestricted assignment-based examinations (ABE) and restricted open-book examinations (OBE) during the COVID-19 pandemic in India: An experimental comparison. Human Behavior and Emerging Technologies, 3(5), 1050–1066. https://doi.org/10.1002/hbe2.290

Şenel, S., & Şenel, H. C. (2021). Use of take-home exam for remote assessment: A case study from Turkey. Journal of Educational Technology and Online Learning, 4(2), 236–255. https://doi.org/10.31681/jetol.912965

Spiegel, T. and Nivette, A. (2023. The relative impact of in-class closed-book versus take-home open-book examination type on academic performance, student knowledge retention and wellbeing. Assessment & Evaluation in Higher Education, 48(1), 27–40. https://doi.org/10.1080/02602938.2021.2016607

Shoufan, A. (2023). Exploring students' perceptions of ChatGPT: Thematic analysis and follow-up survey. IEEE Access.

Tikva, C., & Tambouris, E. (2021). A systematic mapping study on teaching and learning Computational Thinking through programming in higher education. Thinking Skills and Creativity, 41, 100849. https://doi.org/10.1016/j.tsc.2021.100849

Venkatesh, V., Morris, M. G., Davis, G. B., & Davis, F. D. (2003). 'User Acceptance of Information Technology: Toward a Unified View'. MIS Quarterly, 27(3), 425-478. https://doi.org/10.1109/ACCESS.2023.3268224

Wang, Y., Li, H., Feng, Y., Jiang, Y., & Liu, Y. (2012). Assessment of programming language learning based on peer code review model: Implementation and experience report. Computers & Education, 59(2), pp. 412–422. https://doi.org/10.1016/j.compedu.2012.01.007

Yan, D. (2023). Impact of ChatGPT on learners in a L2 writing practicum: An exploratory investigation. Education and Information Technologies, 1-25. https://doi.org/10.1007/s10639-023-11742-4

Zielinski, C., Winker, M., Aggarwal, R., Ferris, L., Heinemann, M., Lapeña, J. F., ... & Citrome, L. (2023). Chatbots, ChatGPT, and scholarly manuscripts-WAME recommendations on ChatGPT and chatbots in relation to scholarly publications. Afro-Egyptian Journal of Infectious and Endemic Diseases, 13(1), 75–79. https://doi.org/10.21608/AEJI.2023.282936