INFORMATION SYSTEMS EDUCATION JOURNAL

Volume 23, No. 1
January 2025
ISSN: 1545-679X

In this issue:

iscap
Information Systems & Computing Academic Professionals

The **Information Systems Education Journal** (ISEDJ) is a double-blind peer-reviewed academic journal published by **ISCAP** (Information Systems and Computing Academic Professionals). Publishing frequency is five times per year. The first year of publication was 2003.

ISEDJ is published online (https://isedj.org). Our sister publication, the Proceedings of the ISCAP Conference (https://iscap.us/proceedings) features all papers, abstracts, panels, workshops, and presentations from the conference.

The journal acceptance review process involves a minimum of three double-blind peer reviews, where both the reviewer is not aware of the identities of the authors and the authors are not aware of the identities of the reviewers. The initial reviews happen before the ISCAP conference. All papers, whether award-winners or not, are invited to resubmit for journal consideration after applying feedback from the Conference presentation. Award winning papers are assured of a publication slot; however, all re-submitted papers including award winners are subjected to a second round of three blind peer reviews to improve quality and make final accept/reject decisions. Those papers that are deemed of sufficient quality are accepted for publication in the ISEDJ journal. Currently the target acceptance rate for the journal is under 35%.

Information Systems Education Journal is pleased to be listed in the Cabell's Directory of Publishing Opportunities in Educational Technology and Library Science, in both the electronic and printed editions. Questions should be addressed to the editor at editor@isedj.org or the publisher at publisher@isedj.org. Special thanks to volunteer members of ISCAP who perform the editorial and review processes for ISEDJ.

2024 ISCAP Board of Directors

Jeff Cummings Univ of NC Wilmington President	Amy Connolly James Madison University Vice President	Eric Breimer Siena College Past President
Jennifer Breese Penn State University Director	David Gomillion Texas A&M University Director	Leigh Mutchler James Madison University Director/Secretary
RJ Podeschi Millikin University Director/Treasurer	David Woods Miami University Director	Jeffry Babb West Texas A&M University Director/Curricular Items Chair
Tom Janicki Univ of NC Wilmington Director/Meeting Facilitator	Paul Witman California Lutheran University Director/2024 Conf Chair	Xihui "Paul" Zhang University of North Alabama Director/JISE Editor

Copyright © 2025 by Information Systems and Computing Academic Professionals (ISCAP). Permission to make digital or hard copies of all or part of this journal for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial use. All copies must bear this notice and full citation. Permission from the Editor is required to post to servers, redistribute to lists, or utilize in a for-profit or commercial use. Permission requests should be sent to Paul Witman, Editor, editor@isedj.org.

INFORMATION SYSTEMS EDUCATION JOURNAL

Editors

Paul Witman
Editor
California Lutheran University

Thomas Janicki
Publisher
U of North Carolina Wilmington

Dana Schwieger
Associate Editor
Southeast Missouri State University

Kevin Mentzer
Associate Editor
Nichols College

Ira Goldstein
Teaching Cases & Exercises Co-Editor
Siena College

Michelle Louch
Teaching Cases & Exercises Co-Editor
Duquesne University

Donald Colton
Emeritus Editor
Brigham Young University Hawaii

Jeffry Babb
Emeritus Editor
West Texas A&M University

A Comparison of Generative AI Solutions and Textbook Solutions in an Introductory Programming Course

Ernst Bekkering
bekkerin@nsuok.edu

Patrick Harrington
harringp@nsuok.edu

Mathematics and Computer Science
Northeastern State University
Tahlequah, OK 74464

Abstract

Generative AI has recently gained the ability to generate computer code. This development is bound to affect how computer programming is taught in higher education. We used past programming assignments and solutions for textbook exercises in our introductory programming class to analyze how accurately one of the leading models, ChatGPT, generates solutions. We selected the ChatGPT-4 available through the Bing search engine for our testing. We used a one-tailed test to calculate success percentage of the textbook versus ChatGPT solutions to determine if there was a statistically significant difference. Neither the book nor ChatGPT provided perfect solutions. Analysis of the results showed that the generated code does not always meet the programming requirements, but also that instructions for generative AI coding and for traditional programming can be improved. We conclude with recommendations for incorporating generative AI in programming classes.

Keywords: artificial intelligence, generative AI, ChatGPT, C++, introductory programming courses.

Recommended Citation: Bekkering, T.E., Harrington, P., (2024). If You Want Something Specific, Ask for it Specifically: A Comparison of Generative AI Solutions and Textbook Solutions in an Introductory Programming Course. *Information Systems Education Journal*. V23(n1) pp 4-22. https://doi.org/10.62273/YQWP1758

©2025 ISCAP (Information Systems and Computing Academic Professionals)

A Comparison of Generative AI Solutions and Textbook Solutions in an Introductory Programming Course

Ernst Bekkering and Patrick Harrington

1. INTRODUCTION

When the news reported that chatbots using GPT-3 could write comprehensive answers to test questions, we tried to see how well they did on essay questions from our CS3343 Computer Operating Systems course. The contrast between ChatGPT answers and textbook answers was eerie. Computer answers were well-written and provided good examples in contrast with the official answers (
Figure **1**). We reported this to faculty at our department meeting in January 2023, but did not give it much mind for the remainder of the semester. Then came the news that large companies had started using generative AI in their IT departments.

Companies using generative AI include Google, Microsoft, Meta, Insider, Duolingo, Slack, Snap, Coca Cola, Instacart, SalesForce, and Shopify (CNBC.com, 2023; Yahoo News, 2023). Companies like Code Academy (Johnson, 2023) and Udemy (2023), as well as traditional educators like Vanderbilt University (Classcentral.com, 2023), the University of Michigan (Michigan Online, 2023), and MIT (2023) have started offering course in generative AI.

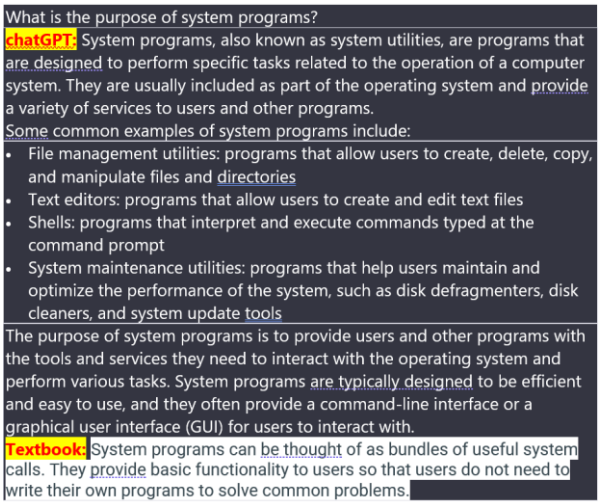

Figure 1:Comparison of answers (personal communication, January 2023)

The revolution in artificial intelligence is now hitting higher education. David Foster, a prominent researcher of AI, states the impending shift as "necessitating a reevaluation and adjustment of current teaching methods and assessment criteria." (Foster, 2022, p. 410).

Because generative AI can write source code, we sought to investigate its limitations and strengths, and begin to determine its usefulness of whether it should be incorporated in teaching Computer Science.

We took the assignments in our Spring 2023 introductory programming course. We compared the results of ChatGPT generated code with our own instructor solutions as posted on the course management system after the assignments were due. Then, we took the end-of-chapter exercise solutions and compared them with the solutions generated by ChatGPT.

The structure of this paper is as follows. We review the literature on generative AI in general and ChatGPT in particular. We briefly review the history of artificial intelligence, describe different types, focus on generative AI, and discuss relevant artificial intelligence in education. Then we describe our methodology in more detail. Following the description of sample and data collection, we analyze the results. Finally, we discuss our conclusions and make recommendations.

2. LITERATURE REVIEW

History of artificial intelligence (AI)

Multiple definitions of AI exist, but a common one is "a system's ability to interpret external data correctly, to learn from such data, and to use those learnings to achieve specific goals and tasks through flexible adaptation" (Kaplan & Haenlein, 2019, p. 17). AI has become a societal focus with the rise of Big Data and increases in computing power (Haenlein & Kaplan, 2019).

Shao et al. (2022) describe the development of AI in three generations. The first generation, Symbolic AI, simulated human intelligence based

on knowledge and experience. Examples are Expert Systems built on knowledge of human experts, natural language processing, financial modeling, and game playing systems. IBM Deep Blue defeated the chess world champion based on raw processing power and analyzing Kasparov's previous games, even though it lacked human understanding of chess strategy.

The second generation (Shao et al., 2022) is data-driven and based on deep learning. New algorithms, such as Convolutional Neural Networks, Recurrent Neural Networks, and Generative Adversarial Networks, emerged. Growth has been accelerated since the models depend on the growth of data without the need for extracting features, and the installed base for storage capacity worldwide is forecast to grow at an annual rate of 19.2% (Statista, 2023). In the future, Shao et al (2022) predict that the third generation of AI will combine knowledge-driven and data-driven theory. Rather than copy brain function, the structure of the brain will be mimicked. This could lead to true Artificial General Intelligence (AGI), but this is in the future. We will now focus more on the development in the second generation of AI since it underlies the current generative AI applications.

AI rapidly changed with the introduction of Deep Learning applied first to the game Go (Silver et al., 2016), followed by a generalization applied to chess and shogi (Silver et al., 2017). This approach is now the basis for applications like virtual assistants, chatbots, entertainment recommendations, humanities compositions, self-driving cars, and fraud detection (Simplilearn.com, 2022). In 2016, DeepMind Applied used Deep Learning to optimize and reduce the energy consumption of its data centers by up to 40% (Evans & Gao, 2016).

The 2017 Google Brain paper, "Attention Is All You Need" describes text generation and conversational AI (Vaswani et al., 2017). It introduced the Transformer Model, which is a neural network architecture that uses attention mechanisms to compute representations of its input and output. Google has made significant contributions with Google Brain (Google, 2023) and TensorFlow (Tensorflow, 2023) as a means for programming convolutional neural networks.

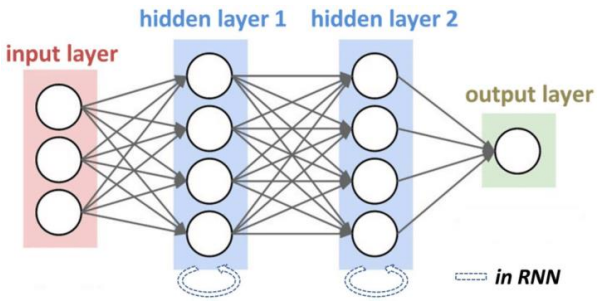

Figure 2:RNN structure (Researchgate.net, 2019)

Before the Transformer Model, recurrent neural networks processed input sequences one element at a time from input layer to hidden layer to output layer (Figure 2). The Transformer Model replaced the fixed internal weights with attention mechanisms to compute the relationships between inputs and outputs (Figure 3).

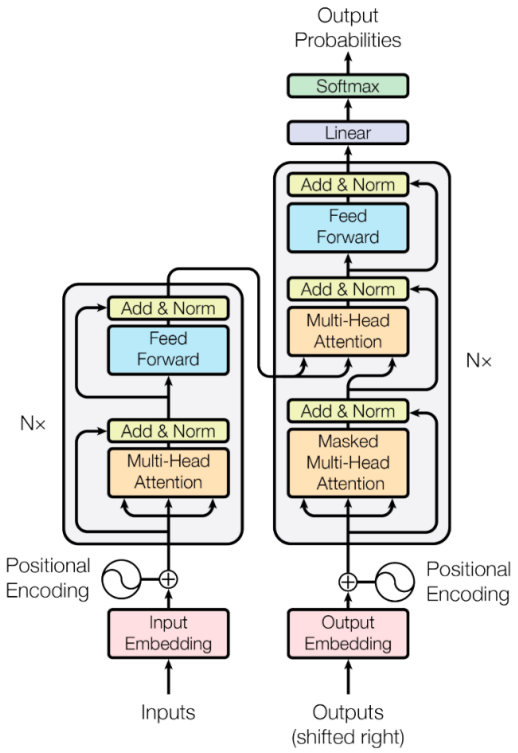

Figure 3: Model architecture of transformer (Vaswani et al., 2017)

In 2018, researchers from OpenAI demonstrated that natural language understanding could be improved by generative pre-training on large amounts of unlabeled text, followed by discriminative fine-tuning for each task (Radford et al., 2018). Researchers from Google reported bidirectional pretraining simplified the fine-tuning phase (Devlin et al., 2018). Other Google

researchers presented a unified framework that addresses text-based language problems and called it T5, "Text-to-Text Transfer Transformer" (Raffel et al., 2020). The wide field of use is illustrated in Figure 4.

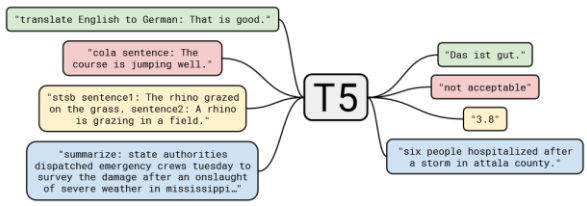

Figure 4: T5 model (Raffel et al., 2020)

Text normalization was improved by augmented use of Batch Normalization (BN), previously used for Computer Vision. Through parameter adjustment, Shen et al. (2020) improved the traditional Layer Normalization (LN) and called it Power Normalization (PN).

Increasing the size of the language model does not necessarily improve the performance. Using human feedback, Ouyang et al. (2022) fine-tuned the performance of GPT-3 in the InstructGPT model, which showed improved truthfulness and reduced toxic output generation with minimal performance reduction despite using 100x fewer parameters in the model.

Types of artificial intelligence

Machine learning includes three types of learning: supervised, unsupervised, and reinforcement learning (Brown, 2021). Supervised learning presents the model with a large data set with example inputs labeled according to the desired output or result. After training, the model can predict outputs in response to new inputs. Unsupervised learning uses large amounts of data without specifying the outcomes. The model produces groupings of sufficient similarity. In reinforcement learning, AI observes and records responses to its actions generated by a simulator running large numbers of cases and evaluates responses using a reward function.

Generative AI is a broad term for AI systems designed to generate content in multiple forms. Images can be generated with websites like Midjourney (2023) or Stable Diffusion (2023). Audio generators include VALL-E (Microsoft, 2023) and resemble.ai (2023). Large Language Models (LLMs) work with language. GPT-4 is the most prominent example, but other models exist (Table 1). Whether code generation tools like Github's Copilot (Github Inc., 2023) and TabNine (TabNine, 2023) should be considered a separate category of content is up for debate because they are computer language tools. Code generation tools offer special features such as code completion, review, and documentation (Tech Point Magazine, 2023).

LLM	Company
GPT-4	OpenAI
Bloom	Hugging Face
AlexaTM	Amazon
ESMFold	Meta AI
Gato	DeepMind
WuDao	Beijing Academy of Artificial Intelligence
LLaMa	Meta
MT-NLG	Nvidia and Microsoft
LaMDA	Google
PanGu-Σ	Huawei
PaLM-2	Google

Table 1: Selected language models

Chat programs

In the area of generative AI, chatbots are special programs that respond to human language in a contextually relevant way. They adapt over time and provide nuanced responses. The programs have the same objective function: "Given a sequence of text, guess what comes next" (Roose, 2023). The best-known example is Chat Generative Pretrained Transformer (ChatGPT).

ChatGPT is a product of OpenAI, a company founded in 2015. Major milestones in its development are (Marr, 2023):

- June 2018: release of GPT-1 with 117 million parameters. It used language understanding tasks for word prediction.
- February 2019: release of GPT-2 with 1.5 billion parameters. It could produce coherent, multi-paragraph text.
- June 2020: release of GPT-3 with 175 billion parameters. It could draft advanced text, answer factual questions, and generate programming code.
- March 2023: release of GPT-4 with 1 trillion parameters. It can use text, video, sound, and image input, output in the same formats, decrease error rates, and is more responsive to user intent (Techradar.com, 2023).

Using LLMs for code generation has not been a deliberate undertaking. As LLMs learn to predict the next word in a sequence, trained over millions or billions of repetitions, they can develop surprising new abilities as emergent behaviors (Mok, 2017). AlphaZero came up with moves such as sacrificing a queen to improve position in chess (Kissinger et al., 2021). In generating the

antibiotic Halicin, new relationships between molecules and lethality to bacteria were discovered (Kissinger et al., 2021).

Generative AI can improve the workplace. In a survey of GitHub developers, 60-75% reported improved work satisfaction, 73% had less effect of context switches, and 87% worked better in repetitive tasks when using Github's AI CoPilot (Kalliamvakou, 2022). Meta evaluated CodeCompose and found that 20% of users reported acceleration of their coding activities, as well as increasing internal and external documentation (Murali et al., 2023).

Use of AI in higher education

Students have flocked to using ChatGPT (Terry, 2023). According to Intelligent (2023), 30% of college students used ChatGPT for schoolwork in the past academic year. Faculty have raised concerns, ranging from cheating and plagiarism, using it to create scholarly work, threats to privacy, fabrication of quotes and references, and lack of trustworthiness (Brandon Paykamian, 2023; Dempere et al., 2023; Freeman-Wong et al., 2023; Lachheb, 2024).

These concerns are not without foundation. ChatGPT is a powerful tool. The new AI tools have now been used to pass multiple exams (Table 2). Educators fear not only that students will use generative AI to create and submit work that is not their own (Kayla Jimenez, 2023), but also that the software may present false, misleading, or ideologically based information. On average, generative AI programs are truthful 25% of the time and absorb underlying social biases from their training data (Stanford University, 2022).

Exam	Percentile score
Uniform bar exam	90th
SAT reading and writing	93rd
SAT math	89th
GRE verbal	99th
GRE quantitative	80th
GRE writing	54th
USA Biology Olympiad	99th

Table 2: GPT-4 exam scores

Use of AI is inevitable, and we should use it in education to help prepare students for a workforce where it will be used (McMurtrie, 2023). Chen (2022) found that machine learning is already at the core of the AI curriculum in the top 46 business schools. Increasingly, experts advocate educational use (Cardona, 2023; Renbarger, 2023).

Furthermore, AI can be used for higher levels of learning (Denny et al., 2023). Students can use AI to create functions with the appropriate sorting algorithm and focus on the structure of the software. Traditional coders could see their job market vanish in a shift to software engineering. As Kissinger et al. (2021, p. 90) explain: "AI coders will complete programs sketched by human developers." Bansal (2024) argues that generative AI will shift the workload from generating code to quality assurance of code.

If higher education does not provide pathways to deep learning, alternative providers will. Cloud services such as Amazon Web Services (AWS) and Microsoft Azure offer pre-built deep learning tools (Amazon, 2023; Azure, 2023). Coursera with DeepLearning.ai are providing a series of online courses. (Coursera, 2023a). IBM offers a similar six course sequence leading to the IBM AI Engineering Professional Certificate (Coursera, 2023b).

Artificial intelligence is like any tool that can be used either for good or for harm. It is the intent and action of the user that matters, not the existence of the tool itself.

3. METHODOLOGY

CS2014 Computer Science I is our introductory programming class. It is taught in C++. The textbook is shared with the follow-up class, CS2163 Computer Science II. CS2014 uses the first 8 chapters of the book; CS2163 uses the rest.

It consists of 3 hours lecture and one hour lab, for a total of 4 credit hours. During the lecture, the instructor demonstrates programs in Visual Studio Code. The programs consist of code in the body of the chapters and the end-of-chapter exercises. The labs use special short exercises with problems and solutions for independent practice. The course has six hands-on programming exercises and a multiple-choice final exam.

The book comes with solutions for the end-of-chapter exercises. We decided to use these solutions to check how well ChatGPT can meet the requirements of these exercises and assignments, as they have been formulated by the textbook author and the class instructor. Since the assignment descriptions are frequently adjusted from semester to semester, we took the most recent instructions from the Spring 2023 semester.

Spring 2023 assignments
The six assignments follow the material presented in the book chapters and focus on specific topics:

- Assignment 1 involves numerical input and sum and average.
- Assignment 2 focuses on loops and output formatting with decimals and tables.
- Assignment 3 introduces file reading, subtotals, and grand totals.
- Assignment 4 uses random number generation, file writing and reading, and nested loops.
- Assignment 5 focuses on functions.
- Assignment 6 works with arrays, sorting, and searching.

The specific descriptions of the assignments are listed in Appendix A.

We used the assignment descriptions to generate ChatGPT instructions and minimized the changes as much as possible. We omitted references to unknown context such as the four-step process (declare variables, assign values, data manipulation, and output or file writing). The ChatGPT instructions are listed in Appendix A next to the assignment instructions for comparison.

Textbook end-of-chapter exercises
Textbooks currently come in paper and electronic format. We used the instructions from the electronic version and made minimal modifications. We had to specify the C++ language. For exercises building on a previous exercise, we copied the instructions from the older exercise and added the modification instructions. Figure 5 gives an example.

We encountered minor problems with incompatibilities between the textbook and exercise solutions. Occasionally the solution numbering was off or no textbook solution was provided, so we matched the solutions with the proper exercise number. If we did not have a textbook solution, we make notes in Appendix B which has the results of the analysis.

Finally, since we did not want to list the textbook instructions with the textbook solutions in this paper, we do not include them in an appendix. They are, however, available upon request to the corresponding author.

Selecting the AI instrument
Multiple tools are currently available for free. We will briefly discuss three of them. All three are web-based rather than software plugins. Students have different preferences for their Integrated Development Environments (IDEs) and copying and pasting from a browser allows them to use their favorite IDE.

First exercise: Random Number Guessing Game Write a program that generates a random number and asks the user to guess what the number is. If the user's guess is higher than the random number, the program should display "Too high, try again." If the user's guess is lower than the random number, the program should display "Too low, try again." The program should use a loop that repeats until the user correctly guesses the random number.
Second exercise: Random Number Guessing Game Enhancement Enhance the program that you wrote for Programming Challenge 20 so it keeps a count of the number of guesses the user makes. When the user correctly guesses the random number, the program should display the number of guesses.
Combined ChatGPT instructions: Random Number Guessing Game Enhancement Write a C++ program that generates a random number and asks the user to guess what the number is. If the user's guess is higher than the random number, the program should display "Too high, try again." If the user's guess is lower than the random number, the program should display "Too low, try again." The program should use a loop that repeats until the user correctly guesses the random number. Enhance the C++ program so it keeps a count of the number of guesses the user makes. When the user correctly guesses the random number, the program should display the number of guesses.

Figure 5 - Combined ChatGPT Instructions

The original ChatGPT is available on the OpenAI website at https://openai.com/chatgpt . It requires setting up an account and logging in. Traffic may be throttled with high use, leading to the error message "ChatGPT has too many requests in 1 hour. Try again later." Since availability to students is a major issue, this disqualified OpenAI for this study.

ChatGPT-4 has been integrated into the Bing search engine on the Microsoft Edge Browser and now also at https://www.bing.com/ (MIT Technology Review, 2023). The Bing chatbot is also plugged into the Bing search engine (Figure

6), so it can get current information from the internet to use in the responses (Tomsguide.com, 2023).

Google Bard is based on the Google LaMDA language model. On April 21, 2023, the CEO of Google announced that Google Bard could generate code including the C++ language. Users must visit the Google Bard page (https://bard.google.com/) and choose "Join the waitlist." Waiting does not exist at the time of this writing. Bard is currently not integrated with a search engine, relies on updates, and can only be used for personal accounts.

We selected the Bing search engine site because it was free, easy to use, and we did not notice any performance issues in our initial testing.

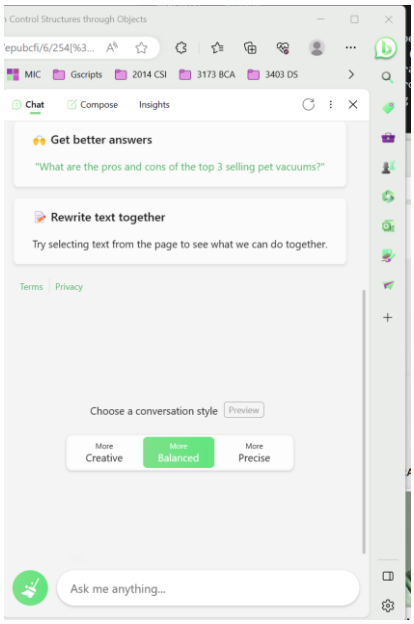

Figure 6: Chatbot in Edge

Using the Bing chatbot

The chatbot is available as part of the Microsoft Edge browser. On each new page, an icon with a "b" opens the chat pane (Figure 6, top right). Users enter their questions and requests in the "Ask me anything" bar and can choose between three conversational styles (Warren, 2023). The default is More Balanced, and we used this style for most of our work. Occasionally, when the chatbot declined to answer and wanted to go to another topic, we used the More Precise style. "More Creative" did not seem appropriate for our work. One reason to go to "More Precise" is the character limit on the ChatGPT input. Whereas More Balanced has a 2,000-character limit, both More Creative and More Balanced have a 4,000-character limit.

4. SAMPLE AND DATA COLLECTION

Legacy assignments

The six course assignments have been discussed in the previous section. Their complete instructions and solutions, and ChatGPT equivalents, are listed in Appendix A.

Textbook exercises

The textbook has eight chapters used for the CS2014 class. The remainder of the chapters are used in the follow-up class, CS 2163 Computer Science II. Table 3 provides an overview of the topics and the number of end-of-chapter exercises.

Scoring assignments and exercises

We copied and pasted the ChatGPT instructions in Appendix A to generate the C++ code for the ChatGPT version. We then copied both the book solutions and ChatGPT code to programs in Visual Studio Code. All programs were scored by the first author on whether (yes/no) all requirements in the instructions were met.

We focused on the explicitly stated requirements as provided in the book description. For instance, textbook authors made extensive use of constants. We only considered the presence or absence of constants when it was specifically mentioned. As another example, chapter 6 on functions preceded chapter 8 on searching and sorting. We did not penalize ChatGPT unless separate functions were specifically mentioned (which they usually were not). The only exception we made was in chapter 2 where the book solutions ran without output. We did consider that an (unspoken) requirement.

5. ANALYSIS AND FINDINGS

Legacy course assignments

ChatGPT was able to meet most requirements.

Assignment 1 was the simplest, with identical source code answers. Both versions met all requirements.

Assignment 2 was more demanding, but standard for repetition and output formatting. Both versions met all requirements.

Assignment 3 used file input to generate a table with annual and grand totals. ChatGPT added arrays, which had not yet been covered. In the Spring 2023 class, this issue was addressed with the general instruction to only use material previously covered. This context was not available to ChatGPT. Both versions met all

requirements. It is a good example of needing to use specific instructions to get specific results.

#	Name	Focus	ex.
2	Introduction to C++	Output #include variables numerical data types C++ strings Operators	20
3	Expressions and Interactivity	Mathematical operations type casting assignment output formatting string class	25
4	Making Decisions	If , if/else, if/else if/else logical operators input checking switch	28
5	Loops and Files	(do) while loops for loops (sub) totals nested loops intro to files	27
6	Functions	Functions (definition, calls, prototypes) return values static variables Reference variables	24
7	Arrays and Vectors	Arrays (definition, accessing, processing) range-based for loop multi-dimensional arrays STL vectors	21
8	Searching and Sorting Arrays	Bubble sort insertion sort linear search binary search	12

Table 3: Textbook Chapters

Assignment 4 is where ChatGPT first failed. The program was to generate a user-specified number of random numbers, to read and calculate on the first 50 numbers, and then to read and calculate on all numbers in the file. Of course, this required closing the file after reading the first 50 numbers so all numbers could be accessed. ChatGPT did not, so the second table was based on 50 too few numbers. In experimenting with requests to fix this error, ChatGPT came up with innovative solutions including setting the file pointer back to the start of the file, but it did not initially meet the requirements.

Assignment 5 required breaking down seconds to days, hours, and minutes. The calculations were correct, but there was one minor deviation in the output. In the original instructions, an example of output was given where 0 days would not be displayed. It is again an example of needing explicit instructions.

In assignment 6, where 25 floats were processed in an array, ChatGPT met all requirements. It used the algorithms library to sort the array, but this was not specifically forbidden.

As a last remark, we would like to note that giving sample output helps ChatGPT to generate code with the same look and feel as originally intended. In the assignments, we have included sample output for students to practice proper input and output. In the end-of-chapter exercises, ChatGPT did not have this advantage.

Textbook exercises

Except for chapter 2, where the book solutions omitted the output, the book solutions outscored the ChatGPT results. Appendix B shows the results by exercise, summarized by chapter and overall. Five exercises did not have a book solution, and even though ChatGPT created working solutions, we excluded the pairs from the sample.

Comparing the chapters, ChatGPT had most difficulties in chapter 6 on functions. On further investigation, this was not due to inability to create separate functions. Most failures resulted from lack of input checking and some logic errors.

Overall, neither the book nor ChatGPT provided perfect solutions. Whereas ChatGPT successfully met the stated requirements 78.9% of the time, the book managed only 90.8%. Examples of book solutions not meeting requirements are in Appendix C. With a valid total of 152 exercise pairs, we wanted to determine if there was a statistically significant difference of the two proportions. This is a simple test that can be performed with a Z-test in Excel (Statology, 2019). Since the book success percentage was higher than the ChatGPT success percentage, we used a one-tailed test. The null hypothesis was

$$H_0: p_b <= p_c$$

and the alternative hypothesis

$$H_a: p_b > p_c$$

where p_b is the book proportion and p_c the ChatGPT proportion. We calculated the z-value as

follows: the book proportion p_b is 0.908; the ChatGPT proportion p_c is 0.789. Both sample sizes s_b and s_c are 152. The pooled sample proportion p_s

$$p_s = (p_b * s_b + p_c * s_c)/ (s_b + s_c) = (0.908 * 152 + 0.789 * 152) / (152 + 152) = 0.84868$$

The test statistic z:

$$z = (p_b - p_c) / \sqrt{p_s * (1- p_s) * [(1/ s_b) + (1/ s_c)]} = (0.908 – 0.789) / \sqrt{0.84868* (1- 0.84868) * [(1/ 152) + (1/ 152)]} = 2.88085.$$

The p-value can be found on a Z table or calculated with the Excel formula

$$p = 1 – NORM.S.DIST(z,true) = 0.00198$$

At a value of 0.00198, the null hypothesis is rejected and the alternative hypothesis of the book success percentage being statistically greater, is correct.

In the course of analysis of the data, combined with our experience using AI in the class, we have several recommendations for using generative AI in introductory programming classes.

- Specify the programming language. ChatGPT would generate solutions in the wrong language.
- Require students to explain each line of code to make sure that they understand the whole program (Figure 7).
- Instruct students that they can only use what has been covered in class. For instance, the use of arrays in assignment 4 was superior to repeatedly reading files. However, arrays had not been covered yet.
- Consider specifying what you want the AI to do, as opposed to instructing it not to do. It worked in 2-15, where the instruction "Please don't use loops. We have not covered that yet" resulted in the proper sequence of output. It did not work in the more complicated 2-16 which used a diamond instead of a triangle, and instructing the AI to avoid loops was not successful. Likewise, in 3-13, ChatGPT used constants because it was explicitly instructed to do so. In contrast, in 3-14, tax rates were not used as constants because there was no instruction to do so.
- Unless you ask for something, you may not get it. The AI would not use variables but hard-coded values. We suggest specifying data types like "use doubles for the membership rates."

```
// with shortened inner loop and smart stop when there are no swaps
void sortArray4(int array[], int size)
{
   int temp; // temporary storage for swapping two values
   bool swap; // track if a swap occurs
   int loop = 0; // for counting the loops to measure performance

   do // start the loop
   {
      swap = false; // we have not swapped pairwise values yet
      // the loop gets shorter each time because the sorted area
      // on the right grows
      for (int count = 0; count < (size - 1 - loop); count++)
      {
        // if left greater than right, need to swap left and right
        if (array[count] > array[count + 1])
        {
           temp = array[count]; // store larger left safely in temp
           array[count] = array[count + 1]; // move larger value to left
           array[count + 1] = temp; // move larger value to the right
           swap = true; // we have made a swap
        }
      }
      loop+=1; // we needed yet another loop
   }while(swap); // continue as long as a swap has taken place

}
```

Figure 7: Commenting to demonstrate understanding.

- The input on the website has a maximum of 2000 characters for input. If one exercise is a modification of an earlier exercise, there may be enough room to paste the instructions for the old exercise before the new one. If there is not, consider going to the slower More Precise style with the 4,000-character limit.
- The outcome of the AI cannot be trusted completely. This allows us, and indeed forces us, to introduce the concept of testing much earlier than before. Tests should be dependent on the requirements of the program. For instance, if the program specifies input range checking, this requires additional tests with out-of-bound inputs. Without the requirement, only valid values should be used and clearly incorrect inputs (e.g., negative ages) avoided. Generative AI has also been known to create non-existing data. We saw this in exercise 6-2, where ChatGPT made up an interest rate. This does not mean that testing the book solutions is any less important. In exercises 4-12 and 4-20, ChatGPT got the math correct and the book did not.
- When running the program to test the output, consider using different numbers than the book sample output. We did not find any instances of hard-coded output but is a (remote) possibility. More importantly, numbers from the book don't always appropriately test the program. For instance, prices like $25 could easily overlook lack of output formatting, whereas prices like $24.78 might give additional information without additional tests.
- ChatGPT often gives explanations of how its generated code works. This could be used in

questions like "Where can you see that the input is between 0 and 100?"
- ChatGPT cannot read figures because it is only text-based at this point. It could not do exercise 4-27 because it was based on a figure. Thus, figures could be used in class to discourage or prevent the use of ChatGPT for tests.
- Book instructions often contain formulas that students need to solve an exercise, but that ChatGPT may not need. For instance, formulas for Future Value or Present Value may not need to be given.
- It helps to give sample output as part of the instructions. This automatically led to the inclusion of the setw() function to create columns of the exact same width as the book solution. We recommend monospaced fonts in the code editor to facilitate checking the results. Wording may matter, e.g. "The program should display a report similar to the following" versus "This is what the program should look like to the user."
- When starting another program, begin a new conversation so old instructions do not influence the results. When modifying the results, specify that the current solution must be used. We found words like "Now use …" helpful.
- Using an online engine is dependent on availability of the service. There are times that the system may not be available or runs slowly.

6. CONCLUSIONS AND RECOMMENDATIONS

Even though the book solutions outperformed the ChatGPT solutions, the comparison is imperfect. With more precise instructions, we might have been able to generate solutions that better met the requirements. The other issue is the quality of the code for both versions. Even though it may not be specified, ChatGPT often provides solutions with higher-level or better programming logic. For instance, sorting and searching algorithms have long been formalized. It may simply be enough for students to recognize the algorithm, learn the relative strengths and weaknesses in a course like Data Structures and Algorithms, and learn to use the algorithm library in this course.

Regardless, generative AI is here to stay, and we will need to incorporate it in our programming classes, starting with introductory classes and progressing to more advanced programming classes as the software gains power. The current competition between technology giants like Microsoft, Google, and Amazon will continue to drive advancements. At the same time, the workflow of software engineers is going to be significantly streamlined and automated.

With the current limitations of AI and the expected rapid development, can we expect to use AI in advanced programming classes? We plan to examine this in the follow-up class CS2163 and the Java-based CS3033 Object-Oriented Programming classes.

Future directions

This study only compares literal copies of book instructions for C++ programming exercises. ChatGPT shows success especially in earlier parts of the book. We plan to expand our work by taking textbook solutions and building instructions to ChatGPT from scratch to recreate the book solutions as closely as possible. We expect this to yield valuable information for faculty and students how to specifically instruct ChatGPT to get specific answers.

7.REFERENCES

Amazon. (2023). Deep Learning on AWS. Amazon Web Services, Inc. https://aws.amazon.com/deep-learning/

Azure. (2023). Azure Machine Learning—ML as a Service | Microsoft Azure. https://azure.microsoft.com/en-us/products/machine-learning

Bansal, J. (2024, March 5). Thanks to AI, the coder is no longer king: All hail the QA engineer. Fast Company. https://www.fastcompany.com/91045570/thanks-to-ai-the-coder-is-no-longer-king-all-hail-the-qa-engineer

Brandon Paykamian. (2023, January 25). Higher Ed Reactions to ChatGPT Run the Gamut. GovTech. https://www.govtech.com/education/higher-ed/higher-ed-reactions-to-chatgpt-run-the-gamut

Brown, S. (2021, April 21). Machine learning, explained | MIT Sloan. https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained

Cardona, M. A. (2023). Artificial Intelligence and Future of Teaching and Learning: Insights and Recommendations. U.S. Department of Education, Office of Educational Technology. https://tech.ed.gov/ai-future-of-teaching-and-learning/

Chen, L. (2022). Current and Future Artificial Intelligence (AI) Curriculum in Business School: A Text Mining Analysis. Journal of Information Systems Education, 33(4), 416–426.

Classcentral.com. (2023, May 17). LLM (Large Language Model) | Free Online Courses | Class Central. https://www.classcentral.com/subject/llm

CNBC.com. (2023, March 11). Why ChatGPT and AI are taking over the cold call, according to Salesforce leader. CNBC. https://www.cnbc.com/2023/03/11/why-chatgpt-ai-are-taking-over-the-cold-call-salesforce-leader.html

Coursera. (2023a). Deep Learning. Coursera. https://www.coursera.org/specializations/deep-learning

Coursera. (2023b). IBM AI Engineering. Coursera. https://www.coursera.org/professional-certificates/ai-engineer

Dempere, J., Modugu, K., Hesham, A., & Ramasamy, L. K. (2023). The impact of ChatGPT on higher education. Frontiers in Education, 8. https://doi.org/10.3389/feduc.2023.1206936

Denny, P., Prather, J., Becker, B. A., Finnie-Ansley, J., Hellas, A., Leinonen, J., Luxton-Reilly, A., Reeves, B. N., Santos, E. A., & Sarsa, S. (2023). Computing Education in the Era of Generative AI (arXiv:2306.02608). arXiv. https://doi.org/10.48550/arXiv.2306.02608

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv Preprint arXiv:1810.04805. https://doi.org/10.48550/arXiv.1810.04805

Evans, R., & Gao, J. (2016, July 20). DeepMind AI Reduces Google Data Centre Cooling Bill by 40%. https://www.deepmind.com/blog/deepmind-ai-reduces-google-data-centre-cooling-bill-by-40

Foster, D. (2022). Generative Deep Learning. O'Reilly Media, Inc.

Freeman-Wong, J., Munguia, D., & Mohr, J. J. (2023). Building a Strategy to Harness ChatGPT in Education. California Management Review Insights. https://cmr.berkeley.edu/2023/08/building-a-strategy-to-harness-chatgpt-in-education/

Github Inc. (2023). GitHub Resources. GitHub Resources. https://resources.github.com/copilot-for-business/

Google. (2023). Brain [Corporate]. About the Team. https://research.google/teams/brain

Haenlein, M., & Kaplan, A. (2019). A Brief History of Artificial Intelligence: On the Past, Present, and Future of Artificial Intelligence. California Management Review, 61, 000812561986492. https://doi.org/10.1177/0008125619864925

Intelligent. (2023). One-Third of College Students Used ChatGPT for Schoolwork During the 2022-23 Academic Year. https://www.intelligent.com/one-third-of-college-students-used-chatgpt-for-schoolwork-during-the-2022-23-academic-year/

Johnson, J. (2023, April 11). What Is ChatGPT & Why Should Programmers Care About It? Codecademy Blog. https://www.codecademy.com/resources/blog/what-is-chatgpt/

Kalliamvakou, E. (2022, September 7). Research: Quantifying GitHub Copilot's impact on developer productivity and happiness. The GitHub Blog. https://github.blog/2022-09-07-research-quantifying-github-copilots-impact-on-developer-productivity-and-happiness/

Kaplan, A., & Haenlein, M. (2019). Siri, Siri, in my hand: Who's the fairest in the land? On the interpretations, illustrations, and implications of artificial intelligence. Business Horizons, 62(1), 15–25. https://doi.org/10.1016/j.bushor.2018.08.004

Kayla Jimenez. (2023, January 30). "This shouldn't be a surprise" The education community shares mixed reactions to ChatGPT. USA TODAY. https://www.usatoday.com/story/news/education/2023/01/30/chatgpt-going-banned-teachers-sound-alarm-new-ai-tech/11069593002/

Kissinger, H. A., Schmidt, E., & Huttenlocher, D. (2021). The Age of AI and Our Human Future. Hachette Book Group.

Lachheb, A. (2024, January 15). ChatGPT in Higher Education: Facts and Ideas to Consider | Online Teaching.

https://onlineteaching.umich.edu/articles/chatgpt-in-higher-education-facts-and-ideas-to-consider/

Marr, B. (2023, May 19). A Short History Of ChatGPT: How We Got To Where We Are Today. Forbes. https://www.forbes.com/sites/bernardmarr/2023/05/19/a-short-history-of-chatgpt-how-we-got-to-where-we-are-today/

McMurtrie, B. (2023). ChatGPT Is Everywhere: Love it or hate it, academics can't ignore the already pervasive technology. The Chronicle of Higher Education, 69(15). https://www.chronicle.com/article/chatgpt-is-already-upending-campus-practices-colleges-are-rushing-to-respond

Michigan Online. (2023). ChatGPT Teach-Out. https://online.umich.edu/courses/chatgpt-teach-out/

Microsoft. (2023). VALL-E. VALL-E. https://vall-e.io/

Midjourney.com. (2023). Midjourney Showcase. https://www.midjourney.com/showcase/recent/

MIT Center for Constructive Communication. (2023). MIT MAS.S68! MIT MAS.S68. https://www.mit.edu/~mas.s68/

MIT Technology Review. (2023, March 23). The inside story of how ChatGPT was built from the people who made it. MIT Technology Review. https://www.technologyreview.com/2023/03/03/1069311/inside-story-oral-history-how-chatgpt-built-openai/

Mok, K. (2017, April 4). Identifying Emergent Behaviors of Complex Systems—In Nature and Computers. The New Stack. https://thenewstack.io/identifying-emergent-behaviors-complex-systems-nature-computers/

Murali, V., Maddila, C., Ahmad, I., Bolin, M., Cheng, D., Ghorbani, N., Fernandez, R., & Nagappan, N. (2023). CodeCompose: A Large-Scale Industrial Deployment of AI-assisted Code Authoring. https://doi.org/10.48550/arXiv.2305.12050

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., & Ray, A. (2022). Training language models to follow instructions with human feedback. Advances in Neural Information Processing Systems, 35, 27730–27744. https://doi.org/10.48550/arXiv.2203.02155

Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving language understanding with unsupervised learning. Technical report. OpenAI. https://openai.com/research/language-unsupervised

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. The Journal of Machine Learning Research, 21(1), 5485–5551. https://doi.org/10.48550/arXiv.1910.10683

Renbarger, M. (2023, January 31). Generative AI is coming for the classroom, whether teachers like it or not. Here's why many in education think it should be embraced rather than shunned. Business Insider. https://www.businessinsider.com/education-experts-teachers-generative-ai-chatgpt-classroom-2023-1

Researchgate.net. (2019). Fig. 3. Generalized (recurrent) neural network architecture with two... ResearchGate. https://www.researchgate.net/figure/Generalized-recurrent-neural-network-architecture-with-two-hidden-layers-The-NN_fig3_337881315

resemble.ai. (2023). AI Voice Generator with Text-to-Speech. Resemble AI. https://www.resemble.ai/

Roose, K. (2023, March 28). How Does ChatGPT Really Work? The New York Times. https://www.nytimes.com/2023/03/28/technology/ai-chatbots-chatgpt-bing-bard-llm.html

Shao, Z., Zhao, R., Yuan, S., Ding, M., & Wang, Y. (2022). Tracing the evolution of AI in the past decade and forecasting the emerging trends. Expert Systems with Applications, 118221. https://doi.org/10.1016/j.eswa.2022.118221

Shen, S., Yao, Z., Gholami, A., Mahoney, M., & Keutzer, K. (2020). Powernorm: Rethinking batch normalization in transformers. International Conference on Machine Learning, 8741–8751. https://doi.org/10.48550/arXiv.2003.07845

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I.,

Panneershelvam, V., & Lanctot, M. (2016). Mastering the game of Go with deep neural networks and tree search. Nature, 529(7587), 484–489. https://doi.org/10.1038/nature16961

Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., & Graepel, T. (2017). Mastering chess and shogi by self-play with a general reinforcement learning algorithm. https://doi.org/10.48550/arXiv.1712.01815

Simplilearn.com. (2022). Top 25 Deep Learning Applications Used Across Industries [2022 Edition]. Simplilearn.Com. https://www.simplilearn.com/tutorials/deep-learning-tutorial/deep-learning-applications

stablediffusionweb.com. (2023). Stable Diffusion Online. https://stablediffusionweb.com/

Stanford University. (2022). The AI index report. https://aiindex.stanford.edu/ai-index-report-2022/

Statista. (2023, November 16). Data growth worldwide 2010-2025. Statista. https://www.statista.com/statistics/871513/worldwide-data-created/

Statology. (2019, June 7). How to Perform a Two Proportion Z-Test in Excel. Statology. https://www.statology.org/two-proportion-z-test-excel/

TabNine. (2023). AI Assistant for software developers | Tabnine. https://www.tabnine.com/

Tech Point Magazine. (2023, February 26). How to use ChatGPT to Code. https://techpointmag.com/how-to-use-chatgpt-to-code/

Techradar.com. (2023, March 14). GPT-4 is bringing a massive upgrade to ChatGPT | TechRadar. https://www.techradar.com/news/gpt-4

Tensorflow. (2023). Create production-grade machine learning models with TensorFlow [Corporate]. https://www.tensorflow.org/

Terry, O. K. (2023). I'm a Student. We're Already Using ChatGPT: No professor or software could ever pick up on it. The Chronicle of Higher Education, 69(15). https://www.chronicle.com/article/im-a-student-you-have-no-idea-how-much-were-using-chatgpt

Tomsguide.com. (2023, April 20). 7 best ChatGPT alternatives I've tested. Tom's Guide. https://www.tomsguide.com/features/chatgpt-alternatives

Udemy. (2023). Online Courses—Learn Anything, On Your Schedule. Udemy. https://www.udemy.com/courses/search/

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, \Lukasz, & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 30.

Warren, T. (2023, March 2). Microsoft now lets you change Bing's chatbot personality to be more entertaining. The Verge. https://www.theverge.com/2023/3/2/23621772/microsoft-bing-ai-chatbot-personality-toggle

Yahoo News. (2023, May 24). Here Are All the Companies Using ChatGPT…So Far. Yahoo News. https://news.yahoo.com/companies-using-chatgpt-far-205500883.html

Appendix A – Description of Assignments and ChatGPT Instructions

The appendix has been moved online due to restrictions on the file size of submitted manuscripts.

Link: appendixA.docx

Appendix B – Pairwise Comparison of Book and ChatGPT Solutions

Exercise	ChatGPT	book	Exercise	ChatGPT	book	Exercise	ChatGPT	book	Exercise	ChatGPT	book	Exercise	ChatGPT	book	Exercise	ChatGPT	book	Exercise	ChatGPT	book
2-1	yes	no	3-1	yes	yes	4-1	yes	yes	5-1	yes	yes	6-1	yes	yes	7-1	no	yes	8-1	yes	yes
2-2	yes	no	3-2	yes	yes	4-2	yes	yes	5-2	no	yes	6-2	yes	yes	7-2	yes	yes	8-2	yes	yes
2-3	yes	no	3-3	yes	yes	4-3	yes	yes	5-3	yes	yes	6-3	yes	yes	7-3	yes	yes	8-3	yes	yes
2-4	yes	yes	3-4	yes	yes	4-4	yes	yes	5-4	yes	yes	6-4	yes	yes	7-4	yes	yes	8-4	yes	yes
2-5	yes	yes	3-5	yes	yes	4-5	yes	yes	5-5	yes	no	6-5	yes	yes	7-5	no	yes	8-5	yes	no
2-6	yes	yes	3-6	yes	yes	4-6	yes	yes	5-6	yes	yes	6-6	yes	yes	7-6	yes	yes	8-6	no	yes
2-7	yes	yes	3-7	yes	yes	4-7	no	yes	5-7	yes	yes	6-7	yes	yes	7-7	no	yes	8-7	no	yes
2-8	yes	yes	3-8	yes	yes	4-8	yes	yes	5-8	no	yes	6-8	no	yes	7-8	yes	yes	8-8	yes	yes
2-9	yes	yes	3-9	no	yes	4-9	yes	yes	5-9	yes	yes	6-9	no	yes	7-9	yes	no	8-9	yes	yes
2-10	yes	yes	3-10	yes	yes	4-10	yes	yes	5-10	yes	yes	6-10	yes	yes	7-10	no	yes	8-10	yes	yes
2-11	yes	yes	3-11	yes	yes	4-11	no	yes	5-11	yes	yes	6-11	no	yes	7-11	yes	no	8-11	yes	yes
2-12	yes	no	3-12	yes	yes	4-12	yes	no	5-12	yes	yes	6-12	no	yes	7-12	yes	yes	8-12	*	*
2-13	yes	yes	3-13	yes	yes	4-13	yes	yes	5-13	yes	yes	6-13	yes	yes	7-13	yes	no			
2-14	yes	yes	3-14	yes	yes	4-14	yes	yes	5-14	yes	yes	6-14	no	yes	7-14	yes	yes			
2-15	yes	yes	3-15	yes	yes	4-15	yes	yes	5-15	yes	yes	6-15	no	yes	7-15	yes	yes			
2-16	yes	yes	3-16	yes	yes	4-16	yes	yes	5-16	no	yes	6-16	yes	yes	7-16	no	yes			
2-17	yes	yes	3-17	no	yes	4-17	yes	yes	5-17	yes	yes	6-17	yes	no	7-17	yes	yes			
2-18	yes	yes	3-18	yes	yes	4-18	yes	yes	5-18	no	yes	6-18	no	yes	7-18	yes	yes			
2-19	yes	yes	3-19	yes	yes	4-19	yes	yes	5-19	yes	yes	6-19	no	yes	7-19	*	*			
2-20	no	no	3-20	yes	yes	4-20	yes	no	5-20	yes	yes	6-20	no	yes	7-20	*	*			
			3-21	yes	yes	4-21	yes	yes	5-21	yes	yes	6-21	yes	yes	7-21	yes	yes			
			3-22	yes	yes	4-22	yes	yes	5-22	yes	yes	6-22	no	yes						
			3-23	yes	yes	4-23	yes	yes	5-23	yes	yes	6-23	yes	yes						
			3-24	yes	yes	4-24	no	yes	5-24	yes	yes	6-24	no	yes						
			3-25	no	yes	4-25	yes	no	5-25	no	yes									
						4-26	yes	yes	5-26	*	*									
						4-27	no	yes	5-27	*	*									
						4-28	no	yes												
ch02			ch03			ch04			ch05			ch06			ch07			ch08		
ChatGPT	19	95.0%	ChatGPT	22	88.0%	ChatGPT	23	82.1%	ChatGPT	20	80.0%	ChatGPT	13	54.2%	ChatGPT	14	73.7%	ChatGPT	9	81.8%
book	15	75.0%	book	25	100.0%	book	25	89.3%	book	24	96.0%	book	23	95.8%	book	16	84.2%	book	10	90.9%
count	20		count	25		count	28		count	25		count	24		count	19		count	11	

*	no solution in the book solution bank	ChatGPT total:	120	ChatGPT %:	78.9%
		book total:	138	book %:	90.8%
		count total:	152		

Legend	
yes	requirements met
no	requirements not met

Two sample -test (one-tailed)		source: https://www.statology.org/two-proportion-z-test-excel/
book proportion	0.908	
book sample size	152	
ChatGPT proportion	0.789	
ChatGPT sample size	152	

Pooled sample proportion	0.84868	=(D40*D41+D42*D43)/(D41+D43)
Test statistic	2.88085	=(D40-D42)/SQRT(E45*(1-E45)*((1/D41)+(1/D43)))
p-value	0.00198	=(1-NORM.S.DIST(E46, TRUE))

Appendix C – Examples of Incomplete Textbook Solutions

Example 1

Sum of Two Numbers
Write a program that stores the integers 50 and 100 in variables, and stores the sum of these two in a variable named total.

```
// Chapter 2, Programming Challenge 1: Sum of Two Numbers
int main()
{
	// Store the integers 50 and 100 in num1 and num2.
	int num1 = 50, num2 = 100;

	// Store the sum of num1 and num2 in total.
	int total = num1 + num2;
	return 0;
}
```

No output - the book forgot to use a cout statement.

Example 2

Software Sales
A software company sells a package that retails for $99. Quantity discounts are given according to the following table.

Quantity	Discount
10–19	20%
20–49	30%
50–99	40%
100 or more	50%

Write a program that asks for the number of units sold and computes the total cost of the purchase.
Input Validation: Make sure the number of units is greater than 0.

```
// Chapter 4, Programming Challenge 12: Software Sales
#include <iostream>
#include <iomanip>
using namespace std;

int main()
{
	// Constant for the unit price.
	const double UNIT_PRICE = 99.0;

	int unitsSold;	// Number of units sold
	double discountPct;  // Discount percentage
	double discountCost; // Unit cost after discount
	double totalCost;    // Total cost

	// Get the number of units sold.
	cout << "How many units were sold? ";
	cin  >> unitsSold;

	// Make sure a positive number was entered.
	if (unitsSold <= 0)
		cout << "Units sold must be greater than zero.\n";

	// Determine the discount percentage.
	else
	{
		if (unitsSold < 10)
			discountPct = 0.00;
		else if (unitsSold >= 10 && unitsSold <= 19)
			discountPct = 0.20;
		else if (unitsSold >= 20 && unitsSold <= 49)
			discountPct = 0.30;
		else if (unitsSold >= 50 && unitsSold <= 99)
			discountPct = 0.40;
		else    // unitsSold was 100 or more
			discountPct = 0.50;

	// Calculate the unit cost after the discount.
	discountCost = UNIT_PRICE * discountPct;

	// Calculate total cost.
	totalCost = unitsSold * discountCost;

	// Display the total cost.
	cout << fixed << showpoint << setprecision(2);
	cout << "The total cost of the purchase is $"
```

```
            << totalCost << endl;
        }

        return 0;
}
```

Output:
How many units were sold? -5
Units sold must be greater than zero.

How many units were sold? 55
The total cost of the purchase is $2178.00

Problem: the book solution is wrong. It gives a 60% discount.

Example 3

```
void getEmployeeInfo(long emp[], int hrs[], double rate[], double pay[], int size)
{
	cout << "Enter the requested information "
		<< "for each employee.\n";

	// Get the information for each employee.
	for (int count = 0; count < size; count++)
	{
		cout << "\nEmployee #" << emp[count] << endl;

	// Get this employee's hours worked.
	cout << "\tHours worked: ";
	cin  >> hrs[count];

	// Validate hours worked.
	while (hrs < 0)
	{
		cout << "Hours worked
		must be 0 or more. "
			<< "Please re-enter: ";
		cin  >> hrs[count];
	}

	// Get this employee's pay rate.
	cout << "\tPay rate: $";
	cin  >> rate[count];

	// Validate the pay rate.
	while (rate[count] < 15.00)
	{
		cout << "Pay rate must be 15.00 or more. "
			<< "Please re-enter: $";
		cin  >> rate[count];
	}

	// Calculate this employee's gross pay.
	pay[count] = hrs[count] * rate[count];
	}
}
```

The program crashes here because it should be hrs[count]