# Examining Student Use of AI in CS1 and CS2*

Eric D. Manley, Timothy Urness,
Andrei Migunov, and Md. Alimoor Reza
Department of Mathematics and Computer Science
Drake University
Des Moines, IA 50311
{eric.manley,timothy.urness,andrei.migunov,md.reza}@drake.edu

**Abstract**

The launch of ChatGPT in November 2022 marked a seismic disruption to many disciplines and industries, including higher education. For the first time, students everywhere have widely available access to a Large Language Model (LLM) capable of generating content - including solutions to programming assignments in CS1 and CS2 - that can pass as the work of a high-achieving student while making traditional plagiarism-detection obsolete. This has spurred various responses in higher education, including a shift to more in-class and unplugged assessments. At the same time, LLMs are transforming the way that many people work, including professional software developers, and students similarly might be able to use them to enhance their learning. In this paper, we report on our experiences with a permissive policy towards the use of ChatGPT and other artificial intelligence (AI) tools for assisting students with their programming assignments in CS1 and CS2 courses in the Spring 2023 semester. Students were allowed to use these tools however they wished as long as they submitted a form which included a transcript of their chat and a reflection on what they learned, if anything, through the interaction. We found that students largely approached the AI in positive ways and that they seemed to genuinely learn from the experience. We also document some things that did not go well and that remain challenges to using AI in programming courses, along with our recommendations on how these might be dealt with in the future.

*Copyright ©2024 by the Consortium for Computing Sciences in Colleges. Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the CCSC copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Consortium for Computing Sciences in Colleges. To copy otherwise, or to republish, requires a fee and/or specific permission.

## 1 Introduction

Shortly after the introduction of OpenAI's ChatGPT in November 2022 [8], students and instructors everywhere quickly realized AI's potential to solve exam problems, generate high-quality essays, and write code on par with high-achieving students. The *New York Times* was among many popular media sources to profile the AI plagiarism threat and how higher education was reacting to thwart it [3], noting many examples of schools that are phasing out asynchronous assignments in favor of in-class written and oral assessments. The article also highlights tools that are being developed to detect AI-generated work along with the schools that are eager to use them.

Computer Science educators had perhaps been more aware of the coming danger than the general population. In their paper *The Robots Are Coming: Exploring the Implications of OpenAI Codex on Introductory Programming*, Finnie-Ansley et al. [1] tested OpenAI Codex (a GPT-3 model fine-tuned for code generation) on 23 typical introductory programming tasks found in CS education literature. The vast majority of tasks were successfully solved by the AI, and the others were all correct with the exception of a trivial formatting error. The authors conclude that the technology "could be considered an emergent existential threat to the teaching and learning of introductory programming" [1]. At the same time, professional software developers have been using it as a resource for learning and increasing productivity [11], either as a standalone tool or integrated into developer tools like GitHub Copilot [2] which had been shown to increase developer productivity [9].

This raises an important question for CS educators: *Do LLMs represent an existential threat that we should fight against or should we embrace it as just another evolution in the way students and professionals work?* Before reacting, we first wanted to know more about how students approach the use of these tools and whether they learn anything as a result. To this end, we implemented a permissive policy towards the use of AI in our CS 1 and CS 2 courses in the Spring 2023 semester. Students were allowed to use them however they wished, including possibly to generate complete solutions to programming assignments, as long as they completed a learning reflection to help us understand their experience. The rest of this paper describes what we learned. We will first cover some additional related work and then discuss our implementation details, results, conclusions and recommendations going forward.

## 2 Background and Related Work

Because LLMs have only recently been made widely available to the public, we are not aware of any studies into how programming students themselves

approach these tools for assistance on their own. However, researchers have investigated other questions intersecting the use of LLMs and education. As discussed above, Finnie-Ansley et al. demonstrated Codex's remarkable ability to solve problems typically encountered in introductory programming courses [1]. MacNeil et al. consider how GPT-3 and Codex can be integrated into CS pedagogy - for generating code explanations, generating programming assignments, and generating code for larger software projects [6]. Kazemitabaar et al. performed a controlled experiment that showed students in introductory programming courses had increased performance when using Codex while note degrading performance on later learning assessments [4].

Looking beyond CS, Rudolph et al. [10] explore the challenges and opportunities for ChatGPT in education and note that the concern that ChatGPT threatens traditional written assignments lends for the opportunity for more innovative, effective assessment with the potential to transform education. Rudolph et al. [10], McMurtrie [7], and Sharples [12] also recommend that the new technology be embraced and incorporated into future pedagogy by exploring how to shape and harness the new tools as opposed to stopping students from using them. Suggestions include using flipped learning to emphasize the critical pieces of work that are completed during class and avoiding formulaic assignments.

## 3 Implementation Details

Students enrolled in one CS1 and one CS2 section at our institution (a private midwestern university), during the Spring 2023 semester were presented with a permissive course policy on using ChatGPT and other AI to assist their learning (as long as they filled out an AI learning reflection form) as well as surveys to discover their views about LLMs. With IRB approval for this study, students were asked for consent to publish data obtained from the surveys, student work, and learning reflections. We removed data for students who did not consent and removed identifying information from those who did. We also removed examples of work where the students did not provide enough information to analyze (for example, some students filled out the requested reflection form but did not provide transcripts of their interaction with the AI). We discuss the policy and surveys below, followed by information on the coding scheme we used to summarize student work.

### 3.1 AI Learning Reflection

The learning reflection form asked students to include the entire transcript of their interaction with the AI tool, even parts they didn't use. It also included questions asking students to explain whether/how they used the AI content

as part of their submission and how they checked its accuracy. Finally, they were invited to reflect on their learning with the prompt *"Give some evidence that shows what you learned from using the AI tool for this assignment. For example, this could be a written description showing you can explain the content in question, some new code that applies what you learned to a different problem, a new version of the code that was changed in sufficient ways to better solve the problem, etc."*

### 3.2 Surveys

Near the beginning of the semester, students were given a survey in order to judge their prior familiarity and views about ChatGPT and similar AI. The purpose of this was to test whether educator fears about plagiarism were validated by student views and whether students were likely to approach these tools for positive use cases (like a kind of AI-Teaching-Assistant - for debugging, help with understanding) or negative use cases (like plagiarism).

*1. How do AI Assisted chatbots, like ChatGPT or IBM Watson, make you feel?* (Nervous or scared; Excited; Interested; Indifferent)

*2. What do you think a college's policy on using AI in classes should be?* (Totally allow; Allow in most cases; Ban in most cases; Totally ban)

*3. How often do you think students in courses like this will turn in AI-generated content as if it were their own work?* (Never; Rarely; Sometimes; Often; Always)

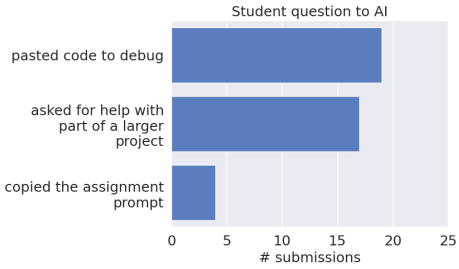
*4: How often do you think students in courses like this one will use ChatGPT to help debug code?* (Never; Rarely; Sometimes; Often; Always)

*5: How often do you think students in courses like this will use ChatGPT to help them understand ideas that they struggle with?* (Never; Rarely; Sometimes; Often; Always)

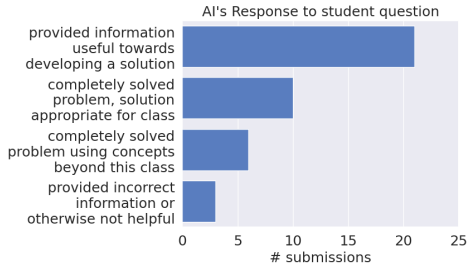
Note that we asked these questions in terms of how often they thought students in courses like this would do these things as a proxy for their own behavior in order to elicit more honest answers - a student might not admit to dishonest intent on the survey, but it might give insight into student perspectives on actions their social circles might find acceptable.

At the end of the semester, students filled out a similar survey which included an additional question on whether they complied with the course policy:

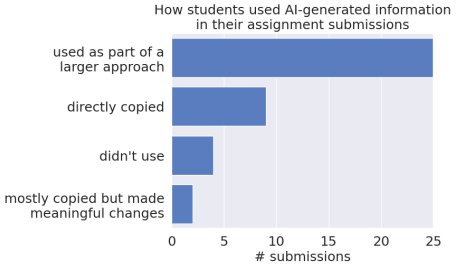
*6. If you used ChatGPT or a similar AI tool to help you with an assignment in this class, how often did you fill out the AI-Assisted Learning Reflection document?* (Not applicable - I didn't use ChatGPT or a similar AI tool for any assignments; Always - I filled out the form every time I used an AI for an assignment; Sometimes - there were times I filled out the form and other times I didn't; Never - I used AI but never filled out the form)



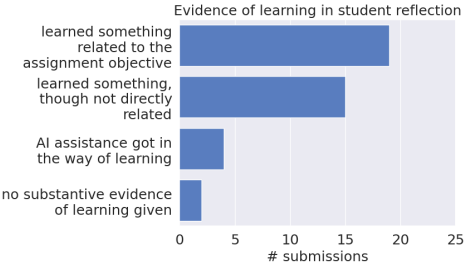

(a) Student Question




(b) AI Response




(c) Student Assignment Submission




(d) Evidence of Learning

Figure 1: Summary of student interaction with AI for assignments

For each of the learning reflections in the study, we reviewed the AI chat transcripts, student learning reflections, and assignment submissions. We categorized them according to four criteria: (1) the kind of question that the student posed to the AI, (2) the kind of response given by the AI, (3) how the student used the information in their submission, and (4) the evidence of learning provided in the student reflection. Fig. 1 summarizes the codes within each criterion along with the number of submissions that exhibited each code.

## 4 Summary of Student Work and Reflections

The study contained 40 submissions, with some students submitting the learning reflection for multiple assignments and others submitting none. As shown in Figure 1a, there were a small number of cases in which the students were seeking a direct solution to the assigned problem. In most cases, students either pasted their own code for debugging or asked for help with only a part of a larger problem. As shown in Figure 1b, the AI usually provided code which either completely solved the problem or was otherwise useful in helping

the students develop a correct solution. Interestingly, in less than a quarter of cases, the students directly copied the code as their solution (see Figure 1c). This is despite the fact that students often found that the AI gave them more help than they were asking for. For example, one student said that they asked the AI

> *"just to get me started/see if there were any basic ideas I was missing. Basically, it did its job "too well" and gave me code that would just finish the assignment."*

Another student mentioned

> *"I wanted to attempt to gradually develop the code with the assistance of chat GPT but the first prompt I gave it, it reuturmed the full code." (sic)*

Students also seemed to genuinely learn important concepts through their interaction with the AI. As shown in Figure 1d, students provided satisfactory evidence of learning in most cases - often centering on the content central to the assignment, but nearly equally often on other important concepts. For example, in an assignment intended to increase proficiency in the use of lists and dictionaries, a student asked ChatGPT about an error caused by trying to access a variable outside of the function it was defined in. The AI gave an explanation which included the following:

> *"The error message you're encounter is because you're trying to call the `most_popular_in_genre` function with the `movies` variable, but `movies` is not defined in your code."*

In fact, the variable was defined *inside* of `most_popular_in_genre` and the ChatGPT suggestions for fixing it were incorrect, but the student was able to fix their code and came away with a better understanding of local vs. global variables and the relationship between arguments and parameters - something that was supposed to have been previously learned. While not the primary intent of the assignment, this was a positive learning experience for the student. In a handful of cases, the AI provided code that got in the way of student learning. For example, in a case where students were supposed to use a custom stack class to solve a problem, the AI solved the problem using a standard list as a stack. The code solved the problem, and the student turned in the code, but it missed one of the main points of the assignment. In another case, the AI solution used a Pandas DataFrame (when the intention was to scan through and filter a list of dictionaries) and included the the operation `filtered_df = df[df["Province"] == selected_state]`, which involves complicated operations that act on the entire Series - something too advanced for students of this level. And yet, the student remarked

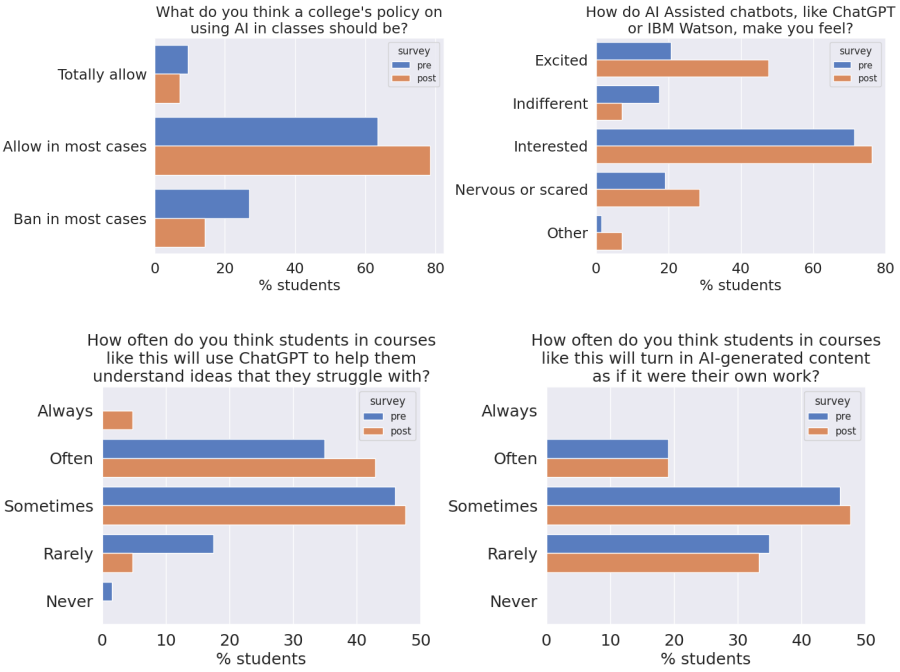



Figure 2: Summary of Student Responses to Survey

> *"I learned about pandas. Pandas has a built-in function called Dataframe that allows you to scan a set of data and makes it much easier to access in the future."*

In this case, we judged that the AI help got in the way of student learning. However, it should be stressed that these were a very small minority of cases.

## 5 Summary of Student Views

Comparing students' responses to survey questions, summarized in Fig. 2, at the beginning and the end of the semester indicates how their views have changed with their experiences with LLMs. One notable difference is their thoughts on the college's policy. At the beginning of the semester, 27% of students selected "Ban in most cases", while at the end of the semester, students seemed to embrace the technology in an academic setting more as only 14% favored banning in most cases.

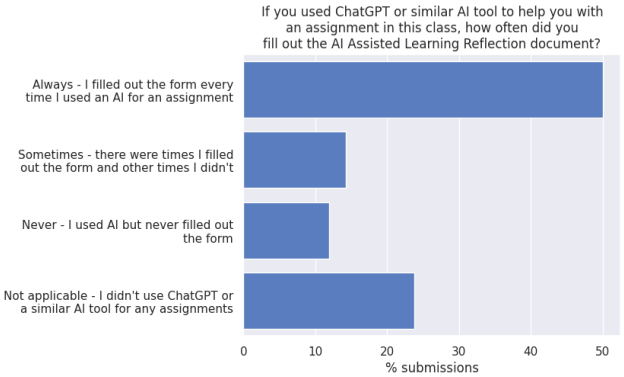



Figure 3: How students reported following the course policy

Another noteworthy difference in the pre and post-semester surveys is the response to the question, "How do AI Assisted chatbots, like ChatGPT or IBM Watson, make you feel?" At the beginning of the semester, only 21% noted "Excited", while at the end of the semester the percentage of "excited" students grew to 48%. Interestingly, there was also an increase in students indicating "Nervous or scared" - perhaps exposure to ChatGPT has made students aware of the potential disruptive nature that these tools introduce to the workforce. Students also seemed to realize positive use cases over the course of the semester, with increases in expected use for understanding and debugging (Q4 data not shown, but saw the "Often" response jump from 30% to 53%).

Lastly, we think it is also remarkable that two thirds of the students thought students would use AI-generated content as if it were their own work at least sometimes, and this perception didn't change much over the course of the semester. When it came to their actual behavior, about a quarter of the students admitted to at least once using AI but not submitting the required form (see Fig. 3). This seems to be in line with the perceptions of some educators who are afraid that the tool will primarily be used in dishonest ways.

## 6 Conclusions and Future Work

Overall, we found that most of the ways that our students used LLMs in our courses were positive and compatible with learning in CS1 and CS2. Most of them prompted the AI to seek help with debugging or with an isolated problem within a larger project that they needed help with. We also discovered some negative outcomes - that in a small number of cases, students used it in a lazy manner, that it is difficult to avoid the AI giving too much help, and that the

AI occasionally gets in the way of learning by presenting solutions above the student's level. Furthermore, some students did not comply with the required course policy to acknowledge and reflect on the assistance they received from the AI - a challenge for teaching students to learn to use AI properly. And, students perceive that their peers are likely to use the technology dishonestly. However, it may be that this is simply a new manifestation of an old problem: convincing students it is in their interest to seek *assistance* rather than *solutions* and to be transparent about sources.

One category of approaches to dealing with these negatives is to *prevent* students from using AI through things like in-class and unplugged assessments. However, for those interested in continuing to explore how AI might enhance student learning, we suggest investigating the following ideas for *mitigating* the negatives:

*Establish an easy way for students to cite AI assistance in their code.* For example, have students link to the transcript from a code comment (which some web-based services now support). By making it easy, we take away barriers to honesty.

*Provide incentives for reflective learning.* For example, give some assignment credit for written reflection on how the activities have led to student learning (whether it involved AI or not) rather than exclusively on the student code, testing, etc.

*Coach students on good ways to use AI.* And then, have them verify it with their citations and reflections - these could be made hard requirements if necessary.

- Have students include their own code/attempts in the AI prompt rather than the assignment text they were given.
- Have students ask follow-up questions on the parts that they do not understand.
- Limit the scope of what they ask of the AI - instead of requesting a solution to the whole problem, they must ask for help with small parts and then integrate solutions into their overall code. This might mean limiting the number of lines of code that can be asked about, but we do not recommend prompting the AI to limit the number of lines of code - in our experience this makes it more likely for it to generate more advanced, though concise, code.
- Explore prompts that tend to lead to more fruitful conversations. The CS education community should explore and share design guidelines for *prompt engineering*, similar to those being investigated in other disciplines [5].

## References


[1] James Finnie-Ansley et al. "The Robots Are Coming: Exploring the Implications of OpenAI Codex on Introductory Programming". In: *Proceedings of the 24th Australasian Computing Education Conference*. ACE '22. Virtual Event, Australia: Association for Computing Machinery, 2022, pp. 10–19. ISBN: 9781450396431.

[2] Nat Friedman. "Introducing GitHub Copilot: your AI pair programmer". In: *GitHub Blog* (June 2021).

[3] Kalley Huang. "Alarmed by A.I. Chatbots, Universities Start Revamping How They Teach". In: *New York Times* (Jan. 2023).

[4] Majeed Kazemitabaar et al. "Studying the effect of AI Code Generators on Supporting Novice Learners in Introductory Programming". In: *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 2023, pp. 1–23.

[5] Vivian Liu and Lydia B Chilton. "Design guidelines for prompt engineering text-to-image generative models". In: *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 2022, pp. 1–23.

[6] Stephen MacNeil et al. "Generating Diverse Code Explanations Using the GPT-3 Large Language Model". In: *Proceedings of the 2022 ACM Conference on International Computing Education Research - Volume 2*. ICER '22. Lugano and Virtual Event, Switzerland: Association for Computing Machinery, 2022, pp. 37–39.

[7] B McMurtrie. "AI and the future of undergraduate writing". In: *The Chronicle of Higher Education* (2022).

[8] OpenAI. *ChatGPT: Optimizing language models for dialogue*. `https://openai.com/blog/chatgpt/`. 2022.

[9] Sida Peng et al. "The impact of AI on developer productivity: Evidence from github copilot". In: *arXiv preprint arXiv:2302.06590* (2023).

[10] Jürgen Rudolph, Samson Tan, and Shannon Tan. "ChatGPT: Bullshit spewer or the end of traditional assessments in higher education?" In: *Journal of Applied Learning and Teaching* 6.1 (2023).

[11] Cliff Saran. "How ChatGPT will become a programmer's best friend". In: *ComputerWeekly.com* (Mar. 2023).

[12] Mike Sharples. "New AI tools that can write student essays require educators to rethink teaching and assessment". In: *Impact of Social Sciences Blog* (2022).