Fostering Creativity: Student-Generative AI Teaming in an Open-Ended CS0 Assignment

Daniel Filcik
daniel.filcik@westpoint.edu
United States Military Academy
West Point, NY, USA

Edward Sobiesk
edward.sobiesk@westpoint.edu
United States Military Academy
West Point, NY, USA

Suzanne J. Matthews
suzanne.matthews@westpoint.edu
United States Military Academy
West Point, NY, USA

Abstract

The increasing ubiquity of web-based generative artificial intelligence technologies necessitates that all students experience teaming with such technologies – exploring their strengths and limitations and learning how to create synergy with them. To aid in this effort, we designed an open-ended generative AI project for the freshmen taking our general-education introduction to computing course. Students were required to team with generative AI to create something beyond what they alone (or the AI alone) could accomplish. Upon completion, students submitted a short written critical analysis documenting their experiences and presented a three-minute demonstration of their project in class. Despite limited course coverage of AI and generative AI prior to this project, we were impressed by the creativity and sophistication of the submitted final products as well as the breadth of generative AI tools explored. Student reflections on the experience illustrated numerous insights into the strengths and limitations of the tools they employed. Our results underscore that students can learn about the benefits and limitations of generative AI in as little as a single assignment and that covering such topics need not require extensive amounts of course time and resources.

CCS Concepts

• **Applied computing** → **Education**; • **Social and professional topics** → **Computing education**; • **Computing methodologies** → **Artificial intelligence**.

Keywords

CS0, Computing Education, Generative Artificial Intelligence, Human-AI Teaming, Final Project, Freshmen

ACM Reference Format:
Daniel Filcik, Edward Sobiesk, and Suzanne J. Matthews. 2025. Fostering Creativity: Student-Generative AI Teaming in an Open-Ended CS0 Assignment. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 1 (SIGCSE TS 2025), February 26-March 1, 2025, Pittsburgh, PA, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3641554.3701853

SIGCSE TS 2025, February 26-March 1, 2025, Pittsburgh, PA, USA

ACM ISBN 979-8-4007-0531-1/25/02
https://doi.org/10.1145/3641554.3701853

1 Introduction

It is a virtual certainty that Artificial Intelligence (AI) will play a central role in the future professional and personal lives of our graduates. Based on this, we argue that *all* students, not just computer science majors, must develop the ability to *team* with AI tools and systems to synergistically accomplish objectives and tasks.

We are entering an indefinite period of transition in which AI will significantly impact and redefine the role and methods of education. Studies have shown that there is a lot of anxiety among faculty involving how students use AI tools [11, 26, 29], with our personal observations being that some disciplines are more reluctant than others to adopt them. Simultaneously, we are seeing curiosity among our students regarding how AI might make their lives more efficient. Additional studies report students are using these tools already in higher education [24, 27], sometimes for unauthorized purposes. We have also observed that students who lack an authorized outlet to employ AI tools either shy away from them or use them surreptitiously. As a result of these quandaries, we were looking for a way to give our freshman CS0 students (and instructors) an outlet to explore generative AI in an *allowed* narrow context and a chance to reflect on what AI can (and cannot) do.

In our required freshman-level introduction to computing course, we sought to realize a Student-Generative AI Teaming goal by introducing all students to the capabilities of generative AI, without designing the entire course around it. We use the word "teaming" because we believe all students must learn to collaborate with AI to solve real-world problems and, in the process, understand the inherent strengths and limitations of such tools. We chose the term "Student-Generative AI teaming" to describe our experience, as we felt the term "Human-Machine Teaming" has too strong an association with the inclusion of robots/autonomous systems [10, 22]. "Human-AI Collaboration" [1, 19] and "Human-AI teaming" [12, 21, 28] seemed closer to our goal. We ultimately adopted the latter term, replacing "Human" with "Student" to reflect the educational context of our work.

We developed an end-of-course project that required students to team with generative AI to create something novel that neither they (nor the AI) could have done by themselves. The created artifact could be software, hardware, a website, artwork, or another project that did not neatly fit into those categories. The goal was for students to demonstrate that they could apply the problem-solving skills and related computing concepts they had learned over the semester within the environment of generative AI. We also wanted them to create something that they were truly excited about and proud of. The project's requirements included a descriptive write-up of their experience and a three-minute in-class demonstration of their final product.

Despite the limited coverage of AI in our course, we were amazed at the diversity and creativity of the projects that students created, with some choosing to work with multiple generative AI tools to produce their projects. Our experience suggests that CS0 students can learn and leverage generative AI without receiving extensive exposure to such topics in class.

2 Related Work

There is a lot of interest today in exploring how students learn with artificial intelligence, but in the computing community, it is mainly focused on how to engage students already interested in computing or computing majors [31]. The literature related to introducing non-majors to AI is limited, with most approaches suggesting AI-specific courses [7, 16, 32] or focusing on teaching students traditional machine learning algorithms or packages [7, 8, 16]. The work of de Freitas and Weingert [8] intersects with our own, as they also teach freshmen non-majors about AI. Unlike their work, however, we did not create an AI version of our existing CS0 course, nor did we focus on *programming* AI systems. Instead, applying principles for working with generative AI similar to those proposed in [20], we introduced students to *generative AI* by requiring them to *team* with such systems to create something novel for a final project.

With the release of web-based generative AI models to the general public, such as OpenAI's text-to-image generator DALL-E in 2021 and the Generative Pre-trained Transformer chatbot (ChatGPT) in 2022, there has been an explosion in both the availability of competing web-based generative AI tools and the interest in studying their impact on computing education. Much of the computing education research looks at the impact of these tools on teaching and learning programming concepts; Cambaz and Zhang have a nice summary [6]. Common topics include using large language models (LLMs) and related generative AI tools to build custom personal tutors for students [5, 17, 18], identifying strengths and weaknesses of such tools in the classroom [13, 30], teaching students prompt engineering [9] and programming [25], and studying student and faculty perceptions of such tools in the classroom [3, 27, 29]. Other papers have covered topics like using LLMs to generate better error messages [14]. Recently, Porter and Zingaro [23] released a book on learning Python using generative AI-assisted tools.

Unlike all of this prior work, our focus is on giving students exposure to the opportunities and limits of generative AI by asking them to team with such tools to create something greater than they (or the tools) could create on their own. Programming was not a requirement, and (as we will discuss in Section 4), several students used generative AI to create products that did not use code.

3 Assignment Details

The United States Military Academy (West Point) is a 4-year undergraduate college in the United States with an enrollment of approximately 4,400 students. During their first two years at West Point, students primarily take courses in the "core curriculum", which is required of all students. CY105 is one such course, with half of the freshman class taking the course in the Fall semester and the other half taking it in the Spring semester. Most of the students who take CY105 have no prior exposure to programming, and nearly all (93%) go on to major in non-computing majors.

3.1 About the course

Computing has been a part of West Point's freshmen core curriculum for 60 years [15], with the expressed goal that every freshman "should have practical exposure to computers, including the writing, running, and testing of computer programs to solve real problems" [2]. The course has gone through several iterations [4], with the modern version of the course consisting of thirty 75-minute lessons. Like a traditional CS0, programming alone is not the sole focus. The course covers a breadth of topics, such as Python programming, computer hardware, networks and the Internet, cybersecurity, copyright, artificial intelligence, and includes a 3-lesson sensor lab focused on using Python to program an Adafruit Bluefruit microcontroller.

Artificial Intelligence is currently covered in only a two-lesson sequence that starts with a discussion about online data collection and transitions to a general coverage of AI, including its current applications and ethical implications. Importantly, we do not have an "AI unit" in the course, given the need to cover a breadth of topics in only thirty lessons. Lastly, the course has no teaching assistants. Instead, the course is taught in parallel every semester in small sections by 12-15 different instructors who administer a common curriculum with the same graded events used for all students.

The traditional version of the final project is an open-ended project that has existed in the course for at least a decade and requires students to demonstrate their programming skills by creating something that fits in one of three topic areas: software (Python only), hardware (Python on their Bluefruit) or a website (HTML/JavaScript/CSS). All project ideas require instructor approval prior to implementation; students are also permitted to choose a project outside the aforementioned topic areas with instructor approval. Once receiving approval, students have roughly three weeks to complete their project. In addition to submitting their files for grading, each student gives a three-minute in-class presentation on their project to their classmates. In Fall of 2023, the project was worth 7.5% of the course grade.

In recent years, instructors have observed that the final projects students were turning in had become increasingly stale. Several CY105 instructors felt that many students did the bare minimum to get an acceptable grade. Similar to [8], we saw too many "cookie-cutter" projects that didn't show much inspiration or creativity.

The widespread availability of generative AI tools in the last two years has forced our institution and our course to institute policies related to their use. West Point gives courses wide latitude on how to incorporate (or avoid) generative AI. Each course is required to lay out an explicit policy on the use of such tools, and each student is required to document their use of any generative AI on out-of-class assignments. While CY105 encourages students to use generative AI as a general study aid, students are not permitted to use AI to generate solutions to homework problems or use it for exams. Beyond this general course policy and a brief mention in our AI lesson, we do not cover generative AI in our in-class content.

3.2 The Generative AI Final Project

Given our desire to give students a "safe space" to explore generative AI, we decided to rewrite the final project to focus on teaming with generative AI rather than pure programming in the Spring

2024 semester. The students were presented with the following assignment description: "The [AI Final Project] is your opportunity to explore and team up with an artificial intelligence tool to create something cool and inspiring. Working with your instructor, identify a project you feel would be exciting to create, and then, working with the AI tool(s) of your choosing, build it (or at least build a working prototype of it)".

We then provided a list of well-known free generative AI tools (and the links to access them) as options to explore. However, students were explicitly told they were not confined to the AI tools on that list. The tools we listed were ChatGPT, Microsoft CoPilot, Replit AI, Gemini, Stable Diffusion, and DALL-E.

Like the previous version of the final project, we asked students to come up with projects in one of several topic areas. Unlike the previous version, we added Artwork as a topic area (thus removing programming as an explicit requirement) and gave specific instructions on what teaming with AI means for each of these areas (shown in Table 1). Importantly, this changed the goal of the project from demonstrating pure programming skills to demonstrating collaborative problem solving with AI. We stressed to students that if the project could be accomplished purely by the AI in just a few prompts, it is likely too simple a project. We also encouraged students to pick a project that "really excites you. This is an opportunity to really stretch your creativity muscles and explore what AI can offer."

Regardless of their project choice, students were required to provide full transcripts of their conversations with AI, screenshots of work in progress, and turn in any source code or other artifacts they created. Students were also required to complete a writeup that reflected on the following questions:

- What did you do? Why is it interesting? What inspired you to pursue this?
- Describe how you used AI to complete your project. Discuss specifically what tools you used, and what prompts you used.
- What parts of the project did the AI complete? Be sure to include a FULL transcript of all the prompts you entered and how the AI(s) responded as an appendix at the end of this document.
- What did YOU specifically do to add to what the AI(s) produced? Provide photos/screenshots of the work in progress as evidence. Remember, if you have source code, the complete source code of your project must be uploaded as part of the zip file you submit for this project.
- If you had more time, what could you do to improve the project further?
- Based on what you learned in the copyright lecture, could you copyright and/or sell your project?
- Based on what you learned, what do you see are some of the potentials and limitations of AI?

Lastly, students were required to give a three-minute presentation of their project during class. Students were graded on the quality of the overall project and presentation, the level of creativity expressed, the strength of their write-ups, and the level of personal contribution to the human-AI team. Specifically, if a student's contribution to the final product seemed trivial (in other words, the AI could do it with minimal prompting), instructors were directed to grade students significantly lower. Lastly, with the new writing component, the final project was increased from 7.5% to 10% of the total grade in the course.

Table 1: Topic Areas for the AI Summative Project

Project Area	Description
Software	Team with AI to write a game, write a puzzle, provide a service, or whatever else you think is interesting. Remember that if AI can generate the source for you (without you having to modify anything), it's too easy. The goal is to team with AI to create something that neither of you could do on your own.
Hardware	Team with AI to write code that can run on your Bluefruit device. This includes finding and modifying the code that programs the hardware. You can also use AI to explore adding additional sensors not covered in the lab. Here are a few sensors we have (your instructor can provide these for you to use): 2-axis Joystick, IR motion finder, passive IR motion sensor, bump sensor.
Website	Team with AI to generate a small website or web-based application. The AI must not generate all elements of the website for you. You will need to learn enough HTML, CSS, and JavaScript to help create a working product.
Artwork	Team with AI to create novel or cool artwork. Be forewarned: you CANNOT rely on the AI for the only artistic expression! If you lack artistic ability, we strongly encourage you to avoid this category, because you will be asked to produce art too that you will be turning into your instructor (with photos of intermediate work), with maximum points given for combining AI with non-digital mediums. Cross-stitch, knitting, sculpture, welding, and woodworking are all valid options!
Other	If you want to do a project outside of these areas, coordinate with your instructor.

3.3 Preparatory Assignment

After we announced the generative AI project to the other instructors teaching CY105 in Spring 2024, many instructors expressed reservations about how successful students would be at producing something of substance, given the lack of coverage of generative AI and prompt engineering in the course.

Partially to assuage instructor concerns, we created a 15-point homework assignment prior to the generative AI project due date that focused on prompt engineering. Modeled after the work of Denny et. al [9], we gave students two "prompt problems" – diagrammatic representations of a problem solving process – and required them to come up with prompts to ChatGPT that produced code capable of passing tests in a provided testing harness. Students were required to turn in links of their ChatGPT conversations and provide a one-paragraph reflection on their experience. Despite

Table 2: Student Choice of Projects By Semester (percentage)

	Software	Hardware	Website	Art	Other
Fall 2023	62.8	8.5	28.7	n/a	0.0
Spring 2024	58.4	9.8	15.0	12.4	4.4

Table 3: Student Choice of Projects By Major and Semester (percentage)

Major	Semester	Software	Hardware	Website	Art	Other
EECS	Fall 2023	57.2	11.4	31.4	n/a	0.0
	Spring 2024	52.2	21.7	21.7	4.4	0.0
Other STEM	Fall 2023	66.7	9.1	24.2	n/a	0.0
	Spring 2024	60.5	11.4	14.1	10.1	3.9
Non-STEM	Fall 2023	58.5	6.9	34.6	n/a	0.0
	Spring 2024	56.2	6.2	15.7	16.2	5.7

this, many instructors remained unconvinced and worried about the quality of projects students would turn in.

4 Preliminary Evaluation

A total 544 students took the course in Spring 2024, representing half of the freshman class, including 11 upperclassmen. In Fall 2023, 526 students took the course, including 450 freshmen (the other half of the class) and 75 upperclassmen. The generative AI project was introduced in Spring 2024; students who took the course in Fall 2023 completed the traditional programming-focused final project.

Prior to data analysis, we sought and received IRB approval from West Point. Quantitative data was derived from our existing course records, while GPA and major information was provided to us through a data request made to our institution. Qualitative feedback was derived from a routine anonymous course-end feedback survey.

4.1 Characteristics of Projects

Did students' choice of a project category change as a result of adding generative AI to the project? Table 2 illustrates how the distribution of project categories changed from Fall 2023 to Spring 2024. Contrary to instructor fears, we did not observe a large drop in the percentage of students who chose projects in the Software category: 62.8% of students in Fall 2023 created a software project, compared to 58.4% in Spring 2024. The biggest change was in the Website category, which 28.7% opted for in Fall 2023 compared to only 15.0% in Spring 2024. Most of that change went to the Art (12.4%) category, causing us to speculate that students who picked the Website category traditionally were ones who were less confident in their programming abilities or looking for creative ways to express themselves. We also observed a slight increase (8.5% to 9.8%) in the number of students choosing hardware projects.

We were curious if a student's eventual major influenced their project choice. Table 3 shows how students in Fall 2023 and Spring 2024 chose projects based on their declared major at the end of freshman year. Students were categorized as being EECS (our department majors, which include our computing and electrical engineering majors), STEM (belonging to one of the other STEM departments at West Point), or Non-STEM. We initially predicted that Non-STEM majors were more likely to create websites; however, roughly 30% of both Non-STEM majors and EECS majors chose to create websites in Fall 2023. In Spring 2024, we observed a marked decrease in the number of students who decided to create websites across all majors. Interestingly, students who would eventually select a STEM major were more likely to pick hardware-related projects than they were in Fall 2023, suggesting that the ability to use generative AI increased their confidence that they would succeed at a hardware-related project. The difference was most striking amongst the students who would eventually declare majors in our department, with the numbers choosing hardware-related projects nearly doubling.

Table 4: Instances of AI Tool Use by Category

	Software	Hardware	Website	Art	Other
ChatGPT	305	50	79	35	19
Repl.it	23	1	11	0	0
Gemini	10	4	5	9	0
CoPilot	8	2	2	7	0
Dall-E	0	1	0	4	0
Stable Diffusion	1	0	0	2	0
Other AI	6	4	4	32	12

How did students team with AI? When we first assigned this project, we fully expected students to pick one tool from the list of generative AI tools that we provided and create a project using that one tool only. We also expected most students would partner with ChatGPT since most students had heard of it previously. While 91% of students did team with ChatGPT, we were surprised to see that a non-trivial number of students teamed with *multiple* AI tools, with an estimated 14% of students teaming with two different AI tools, and 2% teaming with three. Table 4 lists all instances of each AI tool used by project category. Note that since several projects used multiple AI tools, the total is greater than the student population in Spring 2024.

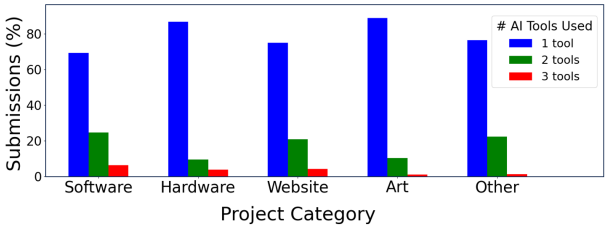

Figure 1: Number of AI Tools Used by Project Category

The number and types of AI tools that students chose to team with varied with the project category they chose. Students who chose projects in the Software, Website, and Other categories were the most likely to use multiple AI tools (Figure 1). In addition, a non-trivial number of students (10.7%) chose to partner with an AI that was *not* on our list of suggestions. This was especially true of the Art and Other categories, where there were 44 instances in which students independently discovered and used AI tools that we did not explicitly suggest (the "other AI tools" category), including

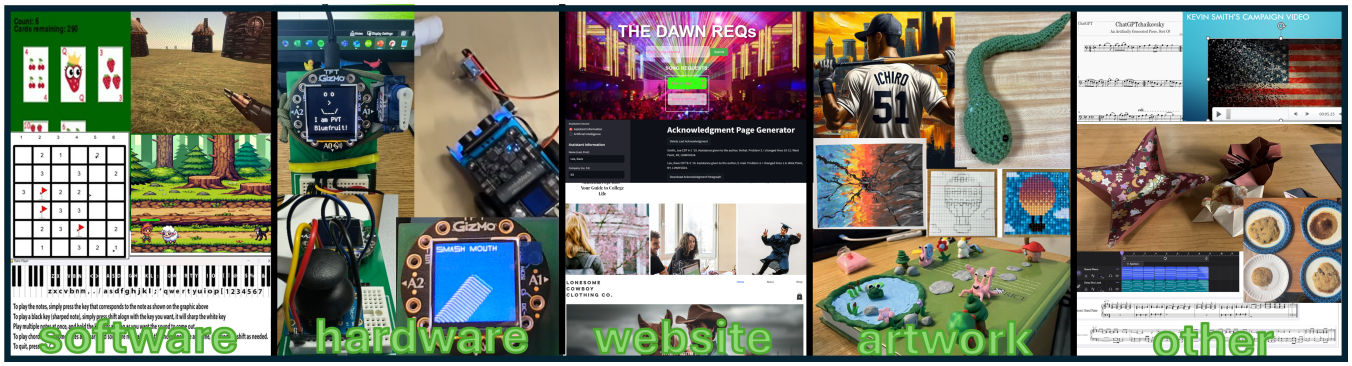

Figure 2: Projects students created by teaming with AI

22 unique tools in Art projects, and 10 unique tools in the Other category.

We were blown away by the richness, diversity, and creativity of the projects that students turned in. Figure 2 shows a very small sampling of what students turned in for each category. For all projects, AI played a key role in helping generate ideas, refine the creative process, and push the limits of what the students could have done without teaming with generative AI.

Despite 58.4% of students choosing to turn in projects in the software category, there was still a diverse variety of (mostly) Python-based projects. Nearly 50% of the software projects focused on games, with several teaming with AI to create novel twists to classic games. The other half largely focused on prediction, tracking, and analysis tools, such as financial calculators, sports analytics, fitness, and flight trackers. These examples illustrate the range of practical and innovative applications that students developed.

In the hardware category, students used generative AI to figure out how to incorporate sensors and generate Python code to program their Bluefruits. Students created games (controllable by buttons or a joystick) and "juke boxes" that played music and displayed cover art. Some students used small educational robots called mBot Neos (lent out by CY105) to create semi-autonomous vehicles that patrolled the classroom or even searched for rats to destroy with a faux-spinning blade (an "eraticator"). Many students reported that generative AI gave them the confidence to try things with the hardware that they would not have done otherwise.

Students who teamed with generative AI in the website category used generative AI to help them write the HTML, CSS, and JavaScript to create very professional websites, despite our course covering next to no web development. The themes of websites varied immensely, from e-commerce sites to web-based games, and even a primitive version of Spotify.

The Art and the Other categories were where students showed perhaps the greatest creativity. Nearly half of the projects in the Art category pursued a non-technology-related interest, such as painting, calligraphy, and crochet. Students used generative AI to help them create designs or patterns for their artwork, and then (after significant iteration with the AI) they executed the AI-generated designs. Some students used generative AI to create digital art projects, such as comic strips, children's books, videos (movie trailers, music videos), and even a campaign video. Others used generative AI to learn or improve their skills (e.g., making origami).

The projects involving music especially caught instructor attention. In these projects (more than 30), students combined their own talents (e.g., playing guitar, singing, mixing) with AI, sometimes augmenting the quality of their own voices, or using AI to come up with interesting lyrics. Other students generated unique drum beats - all taking advantage of the ability to push their musical capabilities in a way many had not tried before this project.

4.2 Feedback from Students and Instructors

The generative AI project was overwhelmingly popular amongst our students. Even though we did not ask an explicit question about the generative AI project in our standard course end survey, 26% (116 of 446) mentioned or discussed the project as the thing they liked most about the course. Among these, 75% (87 of 116) provided expanded feedback, detailing aspects of the project they valued, such as the "freedom" and "creativity" it encouraged. As a representative example, one student said they liked the assignment "because we were given a lot of freedom to choose a topic we were interested in which gave me an incentive to work harder".

Students also expressed appreciation at being given the opportunity to learn about generative AI in a sanctioned manner. Approximately 14% (16 out of 116) of the responses emphasized how the project allowed them to engage with AI responsibly, often for the first time. "The AI project was my favorite part of the course," said one student, "as it allowed me to mix the things we learned in class with a subject matter that I love ... and progressively blend both aspects. Additionally, in a time when AI seemingly has an increasingly substantial role in society, it was refreshing for a class to not just demonize AI but actually teach students how to properly use it for different tasks." Another student shared, "I enjoyed the AI project as it was my first time using ChatGPT for school, as I had always shied away from it because of numerous instances I heard of people misusing it or not properly citing it. However, during and after the project, I saw just how valuable of a tool AI like ChatGPT can be and came to learn how I could possibly use it in the future."

We similarly received positive feedback from the fourteen instructors teaching the course. Despite some initial reservations about the new project, many instructors reported feeling "impressed"

or "pleasantly surprised" with what their students produced. "Forcing students to use generative AI resulted in more creative and complex projects from previous years," reported one instructor who has taught CY105 for over a decade, "I had projects from song writing to crocheting animals. Students used generative AI as a starting point to build something quickly and then were able to expand upon it. They learned they needed to be very specific with the prompts provided to the generative AI to get the intended results."

Not every bit of feedback was positive. In the same standard end-of-course survey, when asked what could be improved or changed about the course, six students (out of 446 respondents) mentioned the generative AI project in some way. One student complained that the project was too "vague," while another complained about having "too much freedom." One instructor, though lauding the use of AI on the final project, felt that students still struggled with prompting the AI correctly. Another instructor felt that "teaming" needed to be better defined in the assignment so that both instructors and students could articulate the concept.

5 Lessons Learned and Conclusions

In this experience report, we describe a CS0-level final project – the generative AI project – that provides students an inspiring experience in teaming with generative AI, exploring its strengths and limitations, and learning how to create synergy with it. Several lessons stand out in the success of this effort, with the following perhaps being the most significant:

1. The generative AI project provides students an injectable introduction to generative AI. Our population of students are required to take our course, and the vast majority (93%) do not pursue computing- or technology-related majors. However, we strongly believe that it is important for all students to learn how they can work with generative AI to create things that matter to them, develop skills to interact with such tools, and understand the opportunities and limitations that the technology currently provides. Our experience shows that the generative AI project succeeds in giving students such an opportunity, without requiring an AI unit or extensive course coverage of AI.

2. Giving students the freedom to select a project they were passionate about was critical to maximizing effort, learning, and creativity. Teaming with generative AI gave students the ability to tackle more challenging problems than they could have accomplished on their own. By enabling students to select a project in a variety of categories (including "no code" topics), we saw an increased pursuit of challenging projects, along with novel uses of AI that we did not predict. Allowing students creative freedom to pursue a project of their interest gave them intrinsic motivation to produce something of high quality.

3. Teaming with generative AI gives students another opportunity to experience complex problem analysis and decomposition. Most CS0 courses emphasize problem solving and decomposition through a variety of short-term (often too superficial) tasks. The generative AI project gives students an opportunity to explore problem solving in a different, longer-term, interdisciplinary context. The iterative nature of interacting with generative AI requires students to clearly articulate their project goals and requirements, decompose their problems in a manner that the AI can understand, and continually refine their inputs to achieve desired outputs. This differs significantly from traditionally structured assignments where instructors articulate problem statements and requirements for students.

4. Teaming with generative AI introduces students to the current power, challenges, and limitations of technologies and humans. Through the AI project, our students learned about the potential for generative AI to augment human creativity, which was a surprising lesson for some, as well as the challenges of managing AI's limitations and even biases. Importantly for freshmen students, this hands-on experience with AI served to demystify the technology, making it more approachable and less intimidating. We argue that non-majors especially need to understand the potentials and limitations of generative AI outside a strictly programming context, and such exposure is crucial for their future professional development. We also believe this project explored and emphasized the importance of the *human contribution* to a human-AI team. Prior to this project, students imagined generative AI to be almost a panacea. However, working with the AI tools illustrated where AI fell short, emphasizing the need to "be the human in the loop" [20].

5. The technology will keep evolving, and we need to be ready. We were very surprised that students discovered and explored 38 AI tools beyond the ones that we had explicitly suggested they explore, which underscores the rapid availability of these tools for a variety of contexts. In this first iteration of the generative AI project, we asked students to reflect on whether the projects they produced were copyrightable. However, there are a number of other ethical, societal, and philosophical concerns students can explore, such as how such tools can (and sometimes do) infringe on existing copyright, displace jobs, affect environmental resources, and are misused for disinformation and cyber-bullying.

The strengths and limitations of generative AI will continue to evolve. There is an ongoing opportunity for students (and instructors) to explore how such technology is a double-edged sword. We believe that assignments like this generative AI project enable educators to practice persistent vigilance in following the rapid evolution of AI and its impact on our students and courses. Generative AI will play a central role in the future professional and personal lives of our graduates, and we believe it would be a moral failure on our part not to continually account for it.

The generative AI final project will be continued in upcoming semesters of the course. We plan on revising the assignment further to ask students to reflect on ethical and philosophical issues and refine our descriptions of what it means to team with AI in an effort to reduce confusion amongst students and instructors. We will also explore how to incorporate AI into other large-scale CS0 assignments and conduct a longitudinal study to track students' abilities and comfort with generative AI as well as the long-term benefits of early AI exposure.

Acknowledgments

The views expressed in this paper are those of the authors and do not reflect the official policy or position of the U.S. Military Academy, the Department of the Army, the Department of Defense, or the U.S. Government.

References

[1] Humaid Al Naqbi, Zied Bahroun, and Vian Ahmed. 2024. Enhancing Work Productivity through Generative Artificial Intelligence: A Comprehensive Literature Review. *Sustainability* 16, 3 (2024), 37. https://doi.org/10.3390/su16031166

[2] Kenneth Alford, Gregory Conti, Eugene Cushen David, Ressler, William Turmel, and Donald J. Welch. 2002. A Two-Edged Sword: Computing At West Point. Making History: West Point at 200 Years Conference.

[3] Matin Amoozadeh, David Daniels, Daye Nam, Aayush Kumar, Stella Chen, Michael Hilton, Sruti Srinivasa Ragavan, and Mohammad Amin Alipour. 2024. Trust in Generative AI among Students: An exploratory study. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1* (Portland, OR, USA) *(SIGCSE 2024)*. Association for Computing Machinery, New York, NY, USA, 67–73. https://doi.org/10.1145/3626252.3630842

[4] Thomas Babbitt, Charles Schooler, and Kyle King. 2019. Punch Cards to Python: A Case Study of a CS0 Core Course. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education* (Minneapolis, MN, USA) *(SIGCSE '19)*. Association for Computing Machinery, New York, NY, USA, 811–817. https://doi.org/10.1145/3287324.3287491

[5] Brett A. Becker, Paul Denny, James Finnie-Ansley, Andrew Luxton-Reilly, James Prather, and Eddie Antonio Santos. 2023. Programming Is Hard - Or at Least It Used to Be: Educational Opportunities and Challenges of AI Code Generation. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1* (Toronto ON, Canada) *(SIGCSE 2023)*. Association for Computing Machinery, New York, NY, USA, 500–506. https://doi.org/10.1145/3545945.3569759

[6] Doga Cambaz and Xiaoling Zhang. 2024. Use of AI-driven Code Generation Models in Teaching and Learning Programming: a Systematic Literature Review. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1* (Portland, OR, USA) *(SIGCSE 2024)*. Association for Computing Machinery, New York, NY, USA, 172–178. https://doi.org/10.1145/3626252.3630958

[7] Andrea Pohoreckyj Danyluk. 2008. Artificial Intelligence for Non-Majors at Multiple Levels.. In *AAAI Spring Symposium: Using AI to Motivate Greater Participation in Computer Science*. Association for the Advancement of Artificial Intelligence, Stanford, CA, 20–25.

[8] Adrian A. de Freitas and Troy B. Weingart. 2021. I'm Going to Learn What?!? Teaching Artificial Intelligence to Freshmen in an Introductory Computer Science Course. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education* (Virtual Event, USA) *(SIGCSE '21)*. Association for Computing Machinery, New York, NY, USA, 198–204. https://doi.org/10.1145/3408877.3432530

[9] Paul Denny, Juho Leinonen, James Prather, Andrew Luxton-Reilly, Thezyrie Amarouche, Brett A. Becker, and Brent N. Reeves. 2024. Prompt Problems: A New Programming Exercise for the Generative AI Era. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1* (Portland, OR, USA) *(SIGCSE 2024)*. Association for Computing Machinery, New York, NY, USA, 296–302. https://doi.org/10.1145/3626252.3630909

[10] Development, Concepts and Doctrine Centre. 2018. *Joint Concept Note 1/18 Human-Machine Teaming*. "UK Ministry of Defense". https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/709359/20180517-concepts_uk_human_machine_teaming_jcn_1_18.pdf Accessed: July 16, 2024.

[11] Veronica Diaz. 2024. Exploring the Opportunities and Challenges with Generative AI. *EDUCAUSE Review* (6 Feb 2024). https://er.educause.edu/articles/2024/2/exploring-the-opportunities-and-challenges-with-generative-ai Accessed: July 17, 2024.

[12] Vera Hagemann, Michèle Rieth, Amrita Suresh, and Frank Kirchner. 2023. Human-AI teams—Challenges for a team-centered AI at work. *Frontiers in Artificial Intelligence* 6 (2023), 8 pages. https://doi.org/10.3389/frai.2023.1252897

[13] Ishika Joshi, Ritvik Budhiraja, Harshal Dev, Jahnvi Kadia, Mohammad Osama Ataullah, Sayan Mitra, Harshal D. Akolekar, and Dhruv Kumar. 2024. ChatGPT in the Classroom: An Analysis of Its Strengths and Weaknesses for Solving Undergraduate Computer Science Questions. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1* (Portland, OR, USA) *(SIGCSE 2024)*. Association for Computing Machinery, New York, NY, USA, 625–631. https://doi.org/10.1145/3626252.3630803

[14] Juho Leinonen, Arto Hellas, Sami Sarsa, Brent Reeves, Paul Denny, James Prather, and Brett A. Becker. 2023. Using Large Language Models to Enhance Programming Error Messages. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1* (Toronto ON, Canada) *(SIGCSE 2023)*. Association for Computing Machinery, New York, NY, USA, 563–569. https://doi.org/10.1145/3545945.3569770

[15] William F. Leubbert. 1965. Computer Science at West Point. *Datamation* 11, 3 (March 1965), 58–67.

[16] Chun-Hung Lin, Chih-Chang Yu, Po-Kang Shih, and Leon Yufeng Wu. 2021. STEM based Artificial Intelligence Learning in General Education for Non-Engineering Undergraduate Students. *Educational Technology & Society* 24, 3 (2021), 224–237. https://www.jstor.org/stable/27032867

[17] Mengqi Liu and Faten M'Hiri. 2024. Beyond Traditional Teaching: Large Language Models as Simulated Teaching Assistants in Computer Science. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1* (Portland, OR, USA) *(SIGCSE 2024)*. Association for Computing Machinery, New York, NY, USA, 743–749. https://doi.org/10.1145/3626252.3630789

[18] Rongxin Liu, Carter Zenke, Charlie Liu, Andrew Holmes, Patrick Thornton, and David J. Malan. 2024. Teaching CS50 with AI: Leveraging Generative Artificial Intelligence in Computer Science Education. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1* (Portland, OR, USA) *(SIGCSE 2024)*. Association for Computing Machinery, New York, NY, USA, 750–756. https://doi.org/10.1145/3626252.3630938

[19] Bernard Marr. 2023. *Boost Your Productivity with Generative AI*. Harvard Business Review. https://hbr.org/2023/06/boost-your-productivity-with-generative-ai Accessed: July 16, 2024.

[20] Ethan Mollick. 2024. *Co-Intelligence: Living and Working with AI* (1st ed.). Portfolio / Penguin, New York, NY, USA.

[21] Dewey Murdick. 2023. *Building Trust in AI: A New Era of Human-Machine Teaming*. Center for Security and Emerging Technology. https://cset.georgetown.edu/article/building-trust-in-ai-a-new-era-of-human-machine-teaming/ Accessed: July 16, 2024.

[22] Tate Nurkin and Julia Siegel. 2023. *Battlefield Applications for Human-Machine Teaming: Demonstrating Value, Experimenting with New Capabilities, and Accelerating Adoption*. "Atlantic Council". https://www.atlanticcouncil.org/wp-content/uploads/2023/08/Battlefield-Applications-for-HMT.pdf Accessed: July 16, 2024.

[23] Leo Porter and Daniel Zingaro. 2023. *Learn AI-assisted Python Programming: With GitHub Copilot and ChatGPT* (1st ed.). Manning, Greenwich, CT, USA.

[24] Arianna Prothero. 2024. *New Data Reveal How Many Students Are Using AI to Cheat*. EducationWeek. https://www.edweek.org/technology/new-data-reveal-how-many-students-are-using-ai-to-cheat/2024/04 Accessed: July 17, 2024.

[25] Laria Reynolds and Kyle McDonell. 2021. Prompt Programming for Large Language Models: Beyond the Few-Shot Paradigm. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI EA '21)*. Association for Computing Machinery, New York, NY, USA, Article 314, 7 pages. https://doi.org/10.1145/3411763.3451760

[26] Molly Roberts. 2023. AI is forcing teachers to confront an existential question. *The Washington Post* (12 Dec 2023). https://www.washingtonpost.com/opinions/2023/12/12/ai-chatgpt-universities-learning/ Accessed: July 17, 2024.

[27] Michael P. Rogers, Hannah Miller Hillberg, and Christopher L. Groves. 2024. Attitudes Towards the Use (and Misuse) of ChatGPT: A Preliminary Study. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1* (Portland, OR, USA) *(SIGCSE 2024)*. Association for Computing Machinery, New York, NY, USA, 1147–1153. https://doi.org/10.1145/3626252.3630784

[28] Jan B. Schmutz, Neal Outland, Sophie Kerstan, Eleni Georganta, and Anna-Sophie Ulfert. 2024. AI-teaming: Redefining collaboration in the digital era. *Current Opinion in Psychology* 58 (2024), 101837. https://doi.org/10.1016/j.copsyc.2024.101837

[29] Judy Sheard, Paul Denny, Arto Hellas, Juho Leinonen, Lauri Malmi, and Simon. 2024. Instructor Perceptions of AI Code Generation Tools - A Multi-Institutional Interview Study. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1* (Portland, OR, USA) *(SIGCSE 2024)*. Association for Computing Machinery, New York, NY, USA, 1223–1229. https://doi.org/10.1145/3626252.3630880

[30] Yiyin Shen, Xinyi Ai, Adalbert Gerald Soosai Raj, Rogers Jeffrey Leo John, and Meenakshi Syamkumar. 2024. Implications of ChatGPT for Data Science Education. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1* (Portland, OR, USA) *(SIGCSE 2024)*. Association for Computing Machinery, New York, NY, USA, 1230–1236. https://doi.org/10.1145/3626252.3630874

[31] Elisabeth Sulmont, Elizabeth Patitsas, and Jeremy R. Cooperstock. 2019. What Is Hard about Teaching Machine Learning to Non-Majors? Insights from Classifying Instructors' Learning Goals. *ACM Transactions on Computing Education (TOCE)* 19, 4, Article 33 (July 2019), 16 pages. https://doi.org/10.1145/3336124

[32] Thomas Way, Lillian Cassel, Paula Matuszek, Mary-Angela Papalaskari, Divya Bonagiri, and Aravinda Gaddam. 2016. Broader and Earlier Access to Machine Learning. In *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education* (Arequipa, Peru) *(ITiCSE '16)*. Association for Computing Machinery, New York, NY, USA, 362. https://doi.org/10.1145/2899415.2925485