Integrating Adversarial Competitions in University Programming Education: An Experience Report

Henry Z Liao, Georgia Institute of Technology

Henry Liao is a Master's student in the Robotics Institute at Carnegie Mellon University. He holds a B.S. in Computer Science from the Georgia Institute of Technology, where he cofounded the educational platform ByteFight to host game AI competitions for computer science classrooms. He now conducts control research for autonomous vehicles at the Driverless Intelligent Vehicle Lab and oversees applications of ByteFight to computer science education at scale.

Tyler Kwok, Georgia Institute of Technology

Tyler Kwok is a software developer in industry. He holds a B.S in Computer Science from the Georgia Institute of Technology, where he cofounded ByteFight. His research focuses on d/Deaf and hard of hearing accessibility and computing education. His work has appeared in NeurIPS and ISWC. He now works on platform infrastructure for large scale systems.

Emmanuel Munoz, Georgia Institute of Technology

Emmanuel Munoz is currently pursuing a B.Sc. degree in computer science at the Georgia Institute of Technology. He is an undergraduate researcher of the People-Agents Research for Computing Education (PARCE) Laboratory, interested in research concerning human-centered computing, computer science education (CS Ed), and human-AI interaction.

Thad Starner, Georgia Institute of Technology

Thad Starner is a Georgia Tech Professor and a Staff Research Scientist at Google Deepmind. Since 1994, Starner has been researching uses of sign language recognition. Professor Starner has been inducted into the CHI Academy in 2017 and AWE's XR Hall of Fame in 2024. In 2025 he became a Fellow in the Association for Computing Machinery and has over 100 issued United States utility patents on wearables, artificial intelligence and interfaces.

Pedro Guillermo Feijóo-García, Georgia Institute of Technology

Dr. Pedro Guillermo Feijóo-García is a Lecturer in the School of Computing Instruction at the College of Computing at the Georgia Institute of Technology. He holds B.Sc. degrees in Systems and Computing Engineering and Mechanical Engineering, and an M.Sc. in Systems and Computing Engineering from Universidad de los Andes. He also earned a Graduate Certificate in University Teaching from Universidad El Bosque and a Ph.D. in Human-Centered Computing from the University of Florida. He directs the People-Agents Research for Computing Education (PARCE) Laboratory, focusing on human-centered computing, computing education, and human-AI interaction, particularly hidden curricula, software engineering education, and AI-supported learning environments. He is a Fulbright alumnus with over 15 years of teaching experience in higher education across Colombia and the United States.

Abrahim Ladha, Georgia Institute of Technology

Abrahim Ladha is a Lecturer in the School of Computing Instruction at the College of Computing at Georgia Institute of Technology, United States. He holds a B.S. in Mathematics and an M.S. in Cybersecurity from Georgia Institute of Technology. He primarily teaches theoretical computer science courses.

©American Society for Engineering Education, 2026

Integrating Adversarial Competitions in University Programming Education: An Experience Report

1 Abstract

Rising enrollment in university computer science (CS) programs has heightened interest in CS education strategies that improve student engagement with course material and accommodate students who desire to innovate upon course concepts. To this end, we propose the idea of integrating competitions as opportunities for learning in university programming education, with the aim of leveraging gamification, competition, and automation to engage students. In this experience report, we present our implementation of an optional adversarial competition as an extension to an online graduate-level artificial intelligence (AI) course with 949 students, of which 174 participated. We design an automated platform with graphical elements and scheduled matchmaking to provide live feedback, help students visualize assignment results, and evaluate their AI's performance with respect to their classmates. After the competition, participants and nonparticipants completed questionnaires to gauge perceptions of learning, satisfaction with the competition, and reasons for participating or not participating. Statistical analyses on responses revealed relationships between time invested in the competition and positive perceptions of learning, as well as between age and participation. This work seeks to apply a live competition to core university programming curriculum at scale, identify traits in both participants' backgrounds and the competition that would lead to increased retention or attrition, and provide an entry point for further exploration into developing integrated competitive co-curricular activities in computing education at scale.

2 Introduction

Enrollment in undergraduate and graduate computer science (CS) programs has continually increased both in schools that offer doctoral programs and in those that do not, with the number of CS students nearly tripled since a decade ago [1], [2]. In response, over the last two decades, much research has examined factors that predict the success of these CS students. These predictors include, but are not limited to: historical indicators such as prior programming experience, prior mathematical skill, or high school test scores [3], [4], academic patterns such as self-efficacy, attribution of success, and study habits [5], [6], and environmental factors such as comfort level, sense of belonging, faculty involvement, and community interactivity [7], [8]. Such a massive influx of students with diverse needs demands effective, creative approaches to course curricula that boost interest in the material and encourage interaction with classmates. This interaction is especially important in light of known issues with retention and engagement in computer science classrooms that stem from a lack of interaction and a lack of motivation [9].

Many strategies for improving student engagement and achievement have emerged in recent years: among the most promising are increasing student involvement in co-curricular activities and introducing gamification of course curriculum. Involvement in co-curricular activities such as hackathons, competitive programming, research, and computing-adjacent interest groups has been shown to improve students' learning skills, raise engagement, increase academic achievement, and boost retention [10], [11]. Gamification and visualization of course material have also been shown to improve CS learning outcomes and student engagement [12]–[14]. Another vector through which to improve CS courses is through efficiency: CS assignments lend themselves to automated solutions that can be applied at scale [15], allowing instructors to manage the increasing number of CS students. This trend can be seen through the widespread proliferation of modern learning management systems (LMS), autograders, and online judge platforms that are used by computer science instructors as instructional and evaluatory tools [16]–[18].

This paper reports on the unification of co-curricular activities, gamification, and automation into a single live-response competitive platform. The platform was used to augment a graduate-level CS assignment with the goal of improving student engagement with advanced course material. We used a post-questionnaire to evaluate reasons for participation or non-participation, perceptions of learning, satisfaction with the competition, and engagement in relation to student demographics and academic variables. The platform was eventually reused in another class for programming education.

Our contributions are as follows:

- Creation of a formalized, reusable, content-agnostic, live-response adversarial competitive platform that utilizes scheduled matchmaking to rank student performance over time.
- Application of our platform at scale to existing core programming curriculum at high frequency.
- Integration of feedback systems in the platform, allowing student iteration via direct, timely, visualized results.

3 Background and Related Work

Student engagement is a challenge in computer science education, particularly as programs struggle to serve large and diverse classrooms [9], [19]. This struggle has driven the development of multiple approaches to improve engagement, each addressing different issues.

Providing timely feedback on assignments is crucial to learning [20]. However, delivering feedback on assignments manually places a substantial burden on teaching staff [21], with programming assignments being particularly difficult to grade by hand [22]. Online judging systems allow teaching staff to provide timely feedback on their programming assignments at scale [18]. Some platforms provide feedback via static code analysis and unit tests, while others seek to provide more nuanced feedback by leveraging machine learning techniques [23]. The success of automated assessment has enabled new approaches to student engagement beyond basic feedback delivery. Research in educational gamification demonstrates that performance tracking, rankings, and social comparison mechanisms can increase student engagement in programming courses [24], [25]. Studies focused on programming language education have shown that gamified environments lead to improved engagement and academic performance [26], [27].

In the context of artificial intelligence (AI), adversarial game environments have emerged as an effective pedagogical approach to enhance engagement and enable the practical application of concepts. Games such as chess and Go have long served as testbeds for AI research [28], [29]. Platforms like Lux AI [30], Terminal [31], and BattleSnake [32] have introduced novel game mechanics with a live leaderboard to serve as competitive educational environments for AI development. One of these platforms, **MIT's BattleCode** [33], influenced many of the design decisions for our platform. These platforms demonstrate the potential of combining automated assessment, gamification, and AI-specific challenges. However, these platforms are typically designed as standalone competitions rather than integrated educational tools, and are closely tied to competition-specific details. How competitive learning platforms can promote student engagement in the classroom has yet to be evaluated. With this in mind, we developed our own platform that maintains full control over data collection for research purposes, ensures implementation simplicity for students, integrates easily with existing assignments, and can be reused to host other educational content easily.

While there have been previous in-class competitions with no live response implemented for university programming education for small pools of ten to 50 competitors [34], [35], [36] and there have been live cocurricular programming competitions [33], there has not been a live, interactive, reusable competitive platform designed to teach core programming content at scale. Our case is the largest application of a live, interactive platform for core university programming curriculum to date, with use by 174 students in the class. The platform was later reused by another undergraduate class of 220 students, where all students participated.

4 Integrating Competition in Class

The main aim of our research was to introduce a live, automated adversarial competition into a university programming classroom and measure student response in terms of engagement. To do so, we collaborated with instructors for the online graduate Artificial Intelligence course at the Georgia Institute of Technology to develop a platform for the class's AI game-playing assignment.

The objective of the class assignment was to design an agent to play a variant of the "Isolation" board game against an opposing AI agent. In the game, two AI agents move pieces around a grid-like board, blocking off cells each time they enter them, such that neither of the agents can re-enter a square that has already been entered. The game is won when the opponent has no legal moves left on their turn. This assignment is used to teach tree search algorithms such as minimax and alpha-beta pruning.

Initially, during the development of the competition, we utilized a placeholder game to test our competition infrastructure. This period included the design of matchmaking between different AI agents, ranking calculations for each agent, and servers to host matches, among other services.

Once we deemed the platform stable enough to deploy, we integrated assignment-specific details into the platform.

4.1 Competition Logistics

Course instructors designed the script that would play two AI agents against each other and return a result as part of the base assignment's autograder. This script served as the competition's game runner. Student submissions were validated by simulating a match environment and evaluating whether student programs could return a valid legal first move. Once a user submitted a valid submission, they would be assigned an Elo rating of 1200 and entered into circulation for matchmaking. A submission that passed the entirety of the class assignment's autograder would always be a valid submission to our competition as well. The platform was designed to be easily reusable via a similar process for future instructors who desired to host similar competitions for their classes. All that would be required is for another instructor to build a script that accepts local submissions and simulates the result, and the platform would be able to host a live competition for it.

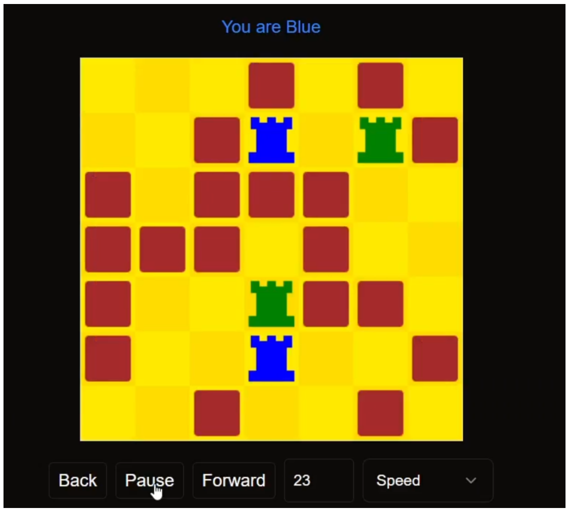
After we showcased the final iteration of the competition platform, course instructors decided that the reward for performing well in the competition would be extra credit points added to students' assignment scores. The top one percent of the final leaderboard would receive ten bonus points, the top five percent would receive six bonus points, and after that, the rest of the top ten percent of submissions would receive three bonus points. One week after the AI game-playing assignment was released to the class, the rules of the extra credit were announced, and our implementation of the extra-credit opportunity was released. When submissions closed two weeks later, 80 rounds of matchmaking were played between students with decreasing Elo k-factor to converge ratings, after which standings were finalized. We compiled the top ten percent of players for course instructors to distribute points to.

Three weeks after the submissions closed and standings were finalized, we announced an optional post-questionnaire available to the entire class. The questionnaire could be completed for 5 points of extra credit added to their assignment score. This questionnaire collected responses on demographics, perceptions of learning, and qualitative questions regarding the competition.

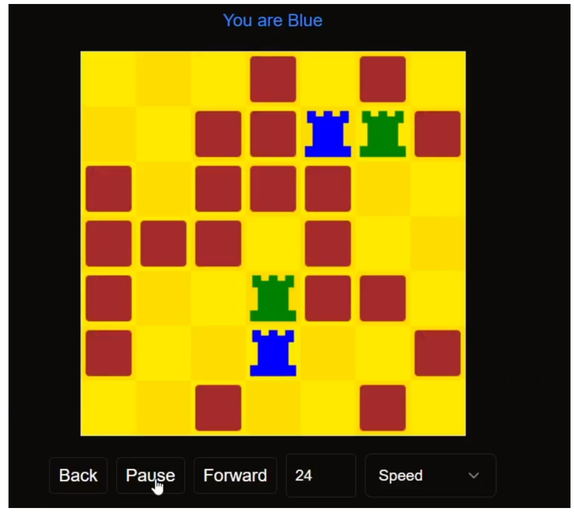
4.1.1 Match visualization

Unlike other in-class programming competitions that provide results only after a full competition has run, we provided a live match replay system that was created to visualize previously played matches so that students could better understand the decisions their AI algorithms were making (see Figure 1). These matches would be available seconds after the match finished playing, allowing students to quickly receive feedback in the same way an autograder would, which provided students the opportunity to quickly iterate once their algorithms had been deployed to play against other students.

This replay system consisted of a single view of the board, which could play through the match as an animation. Alternatively, participants could step through each move of the match by hand for more granular review. Matches were loaded into the visualizer from match histories stored in our database.



(a) Turn 23 of a match



(b) Turn 24 of a match

Figure 1: Web viewer playing back a recorded match between two participants as an animation.

4.2 Questionnaire and Participants

During the two-week period that submissions were open, 174 students submitted valid AI agents to the platform, totaling 507 valid submissions. Of the students that submitted, 122 (70% of 174) submitted only one valid submission. 52 (29%) students submitted multiple valid submissions. Of those 52 students, 34 students submitted two or three submissions, and 18 (10%) submitted more than three valid submissions.

Three weeks after the competition ended, 475 consenting students responded to our post-questionnaire. 19 questionnaire submissions were discarded due to incompleteness and contradictory statements in their responses.

Of the 456 students who submitted valid responses to the questionnaire, 126 responded that they participated in the competition and 330 (72%) responded that they did not. 331 (73%) of the questionnaire responders identified as male, 112 (25%) of the questionnaire responders identified as female, three (1%) of the questionnaire responders identified as nonbinary, and ten elected not to report. Among the responders, 247 (54%) identified as Asian, 160 (35%) identified as White (Non-Hispanic / Non-Latin American), 29 (6%) identified as Hispanic/Latin American, 16 (4%) identified as Middle Eastern/North African, 14 (3%) identified as Black/African American, two identified as Native American/Alaska Native, two identified as Pacific Islander, five identified as other ethnicities, and 17 chose not to report. 219 (48%) of students who responded to the questionnaire were English learners, and 237 (52%) were native English speakers. 340 (75%) of responding students were domestic students, whereas 116 (25%) were international. The highest rates of response for international students came from the following countries: Canada at 25 students (5%), India at 17 students (4%), China at 14 students (3%), Singapore at nine students (2%), South Korea at seven students (2%), United Kingdom of Great Britain and Northern Ireland at 5 students (1%), and Japan at four students (1%). No other countries had more than two questionnaire respondents, and 18 students chose not to identify their nationality. The median age of all respondents was 29.

451 (99%) questionnaire respondents were graduate students, while five (1%) were undergraduate students. 366 (80%) students who responded to the questionnaire were working full time, whereas 90 (20%) were not. The breakdown of students by program year was 207 (45%) in their first year, 125 (27%) in their second year, 66 (14%) in their third year, 35 (8%) in their fourth year, and 22 (5%) in their fifth year or more. The distribution by major was 429 (94%) students studying Computer Science, 18 (4%) studying Data Analytics or Data Science, five (1%) studying Computer Engineering, and four students each studying different other majors.

5 Competition and Platform Design

The development of the architecture (see Figure 2) began in the Fall of 2024 and lasted until February of 2025. Our main focus was the creation of an automated system that could register participants, receive AI game-playing agent submissions, schedule equitable game matches between AIs for all participants, and run the matches on dedicated servers. The guiding ideas behind our design process were: *How do we design a system for many students that de-emphasizes the pressure of course evaluation, encourages self-driven exploration of advanced concepts, and facilitates interaction with classmates?*

5.1 Platform Architecture

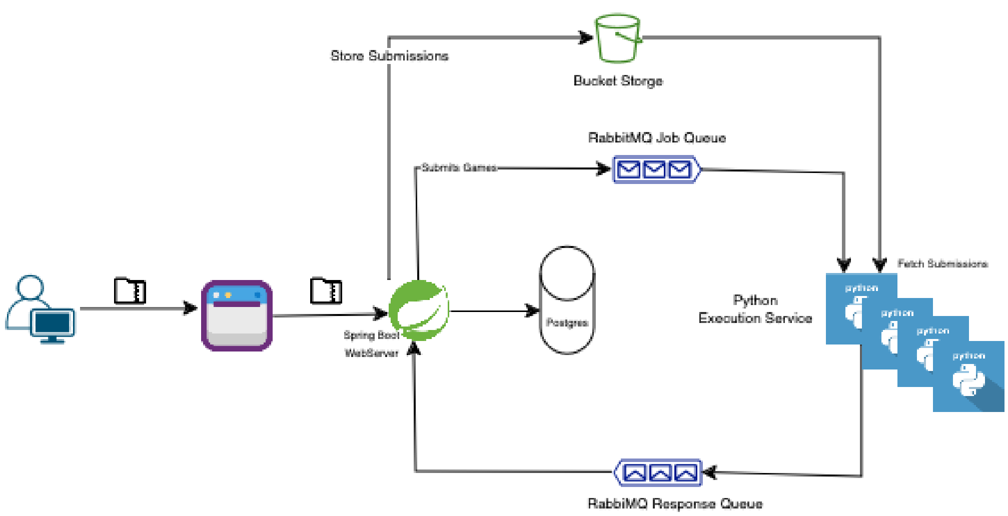


Figure 2: Architecture for the competition.

We decided that for a more streamlined implementation process and to emphasize the optionality of the assignment extension, it was best if students signed up on a dedicated website separate from the course LMS using their names and school-affiliated emails. Identifiable information (such as first name, last name, and email) was kept solely on Supabase, an open-source Postgres development platform. After logging in, participants would submit the AI that they developed for the AI game-playing assignment to the website in the same format that they submitted it for class. Code submissions would be transferred to Google Cloud Platform for bucket storage. No identifiable information was stored on Google Cloud Platform.

We implemented a Spring Boot web server to host our services, which could be accessed via API calls. These services included registration, authentication, scheduling matches between participants, handling match results, and retrieving data from our database. Both the web server and website were deployed on the Railway platform.

We also developed dedicated Python game servers, which could each run a match and return which participant won, which participant lost, and each move that was made by the AIs. Multiple instances of these dedicated game servers were hosted on a single computer with a hyperthreaded six-core 2018 Intel i7 processor provided by course instructors, with each game server capable of processing matches in parallel.

The typical life cycle for a match was as follows: Every hour, the web server would schedule four matches for every player via the construction of four-regular graphs [37]. Match metadata would be submitted to a RabbitMQ Job Queue, where dedicated game servers would be listening. The game servers would proceed to download the relevant AI submissions from storage, run the match, and return the match history on a RabbitMQ Response Queue. The web server would process these matches from the response queue, store match histories in the database, and recalculate the win/loss record and Elo rating for the students involved in the match. Stored match histories could be processed into a visualization by our web visualizer. Per request by course instructors, we did not allow students to play additional matches against each other outside of scheduled matchmaking - this was to prevent students from using the platform as a test environment for the assignment rather than learning to test their programs locally.

5.2 Student Experience

When students signed up, they were placed on their own team with their first and last names as the team identifier. On the student's team page (see Figure 3), they were able to anonymize themselves to other players by changing their team name. Students were also allowed to submit their own "quote" that would be displayed alongside their username on the leaderboard. The team page showed charts that tracked match statistics and Elo rating over time. Individual matches were displayed as a list at the bottom of the individual team page, and each contained a link that allowed the user to review the match in our web visualizer. Rankings, team names, team quotes, and team Elo were publicly displayed on a live leaderboard that updated after every match.

6 Findings and Results

The post-questionnaire was released after the final rankings had been finalized. This questionnaire was optional and available to all students in the class. The incentive for fully completing the questionnaire was five extra credit points added to the assignment category.

The demographic portion of the questionnaire was adapted from prior studies examining similar populations [38]. Questions regarding factors that may have been relevant to students' ability to participate were appended to this part of the questionnaire. Such factors included whether students were working a full-time job, the number of credit hours they were taking, and the number of extracurricular activities they were involved in. The rest of the questionnaire was comprised of Likert scale questions evaluating students' perceptions of learning, free response

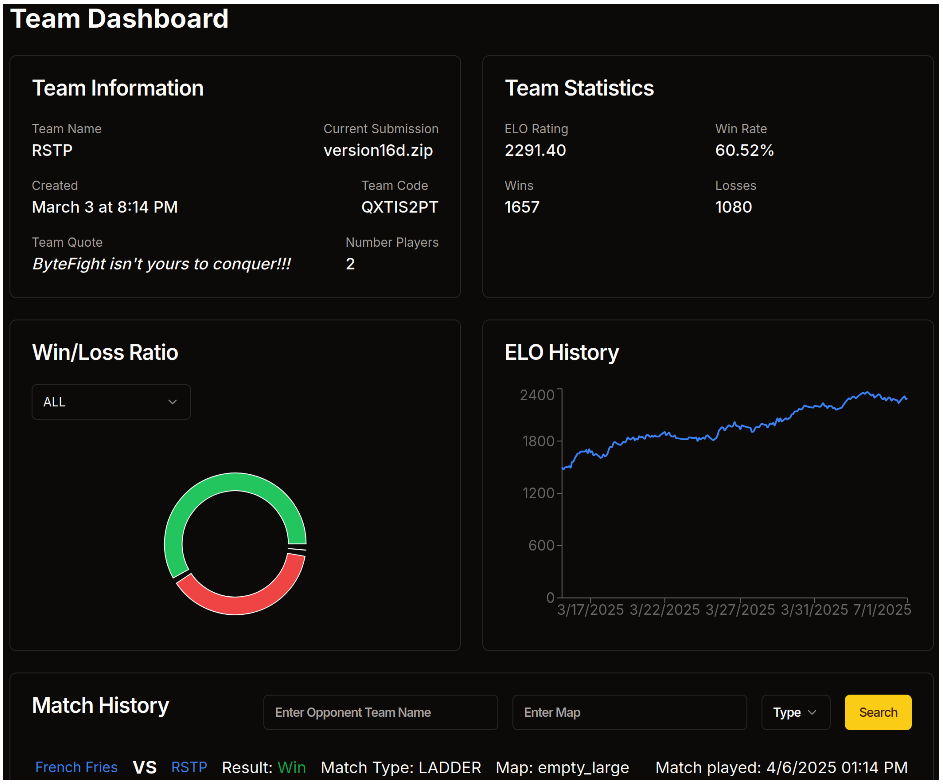


Figure 3: Development version of team page used for viewing competition statistics and matches.

questions gauging students' reasons for liking, disliking, or not participating in the competition, and participants' own estimation of performance and investment in the competition.

It should be noted that due to the optionality of the assignment and the subsequent questionnaire, our results have potential for selection bias, with students who enjoyed the competition and assignment more choosing to engage with the platform and subsequent survey at a higher rate than their peers.

All ratio variables, including age, extracurricular involvement, hours spent in the competition, and current course load, were analyzed for outliers. Outliers were removed from relevant analyses according to the following criteria: $2 \times IQR$ for age (ten of 456 observations) and $1.5 \times IQR$ (three of 456 observations) for course load.

6.1 Perceptions of Learning and Self-Evaluation of Success

Likert scale questions were only presented to students who participated in the competition, which consisted of 126 survey respondents. The first questions dealt primarily with students' perceptions of learning during the competition, and were adapted from Diemer et al. [39]. The last question referred to students' perception of their own performance in the competition. The questions used reverse wording to reduce straightlining and were ordered as follows:

- Please indicate how much you agree with the following statements concerning the extracurricular competition as part of your learning process:

  - (Q29_1) Competing in the competition helped me apply course content to solve problems.
  - (Q29_2) Competing in the competition did not help me learn the course content.
  - (Q29_3) Competing in the competition helped me connect ideas in new ways.
  - (Q29_4) Competing in the competition did not help me participate in the course in ways that enhanced my learning.
  - (Q29_5) Competing in the competition helped me develop confidence in the subject area.
  - (Q29_6) Competing in the competition did not help me develop skills that apply to my academic career and/or professional development.
- (Q35_1) On a scale of 1 (Not well at all) to 7 (Very well), how well do you consider you did in the competition?

Of the students who chose to respond to the survey, sentiments toward perceptions of learning for the competition were on average positive for all relevant questions, regardless of positive or negative wording (see Table 1).

	Q29_1	Q29_2	Q29_3	Q29_4	Q29_5	Q29_6
Mean	5.189	2.721	4.059	2.713	4.885	2.746
Std. Deviation	1.294	1.678	1.402	1.654	1.596	1.639
25th Percentile	4.000	1.000	4.000	1.000	4.000	1.000
50th Percentile	5.000	2.500	5.000	2.000	5.000	2.000
75th Percentile	6.000	4.000	6.000	4.000	6.000	4.000

Table 1: Descriptive statistics for Likert scale data.

6.1.1 Categorical analyses

Due to concerns over whether the Likert scale should be treated as interval or general ordinal data [40], [41], we elected for a conservative approach and used non-parametric methods to analyze our Likert scale questions. We conducted an exploratory analysis of relationships between Likert scale questions and single categorical variables using Kruskal-Wallis tests to determine where best to focus our hypotheses for interplay between variables. All variables identified as promising were used to create pairwise-merged categories for confirmatory testing if possible. Kruskal-Wallis tests were then conducted to examine differences in Likert-scale responses across these merged groups. No meaningful correlations were found; hence, there was no need for post-hoc testing.

6.1.2 Correlational analyses

We used Spearman's rho for correlational analysis due to the assumed non-parametric nature of Likert scale data. Results indicated a correlation between time spent on the competition by survey participants and Likert scale questions (Q29_1), (Q29_2), (Q29_3), (Q29_4), (Q29_5), and

(Q29_6), all with weak to moderate correlation strength (see Table 2). Positive correlations are observed for statements with positive wording regarding the competition, and negative correlations are observed for statements with negative wording regarding the competition, except for (Q29_3). Another set of Spearman's rho correlations were done to examine the relationship between the age of participants and Likert-scale questions; no correlations were found.

Variable		Q29_1	Q29_2	Q29_3	Q29_4	Q29_5	Q29_6	Q35_1
Age	Spearman's ρ	0.126	-0.011	0.230*	-0.153	0.145	-0.078	0.105
	p-value	0.166	0.905	0.011	0.092	0.110	0.384	0.249
Hours spent	Spearman's ρ	0.372***	-0.341***	-0.455***	-0.277**	0.319***	-0.369***	0.209*
	p-value	< 0.001	< 0.001	< 0.001	0.002	< 0.001	< 0.001	0.019

$^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$

Table 2: Correlations between Likert-scale questions and age and time invested in competition.

6.2 Participation and Time Investment

When investigating participation and time investment, we first examined their relationship to categorical student demographics. Chi-square tests for independence on relationships between individual categorical variables and participation in the competition did not yield any significant associations. Kruskal-Wallis tests on relationships between individual categorical variables and time invested in the competition also did not indicate any significant differences in time invested across these categories.

We proceeded to examine the relationship between ratio variables, including age, extracurricular involvement, and course load, with participation and time invested in the competition.

Mann-Whitney U-Tests were conducted on the relationship between ratio variables and participation in the competition. Results indicated a potential age difference between participating and non-participating survey respondents (Mann-Whitney U-Test, $H(1) = 16264$, $p < .005$). Further inspection revealed that the age of survey respondents who competed in the competition was, per quartile, uniformly two years older than those who did not compete in the competition.

Correlational tests between the ratio variables and time invested indicated a correlation between age of survey respondents and time invested, with a moderate positive magnitude (see Table 3).

Variable		Hours spent	Age
Hours Spent	Spearman's ρ	—	
	p-value	—	
Age	Spearman's ρ	0.319***	—
	p-value	< 0.001	—

$^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$

Table 3: Correlational analysis between age and hours spent per week in the competition.

Other correlational analyses between ratio variables and time invested did not yield any significant relationships.

6.3 Free Response Analysis

Free-response questions were a qualitative measure of general reasons for satisfaction or dissatisfaction with the competition. Our free-response question for students who did not participate in the competition was:

- (Q57) Why did you decide not to participate in the competition?

Our free-response questions for students who participated in the competition were:

- (Q30) Tell us what you liked about the extra-curricular competition to learn course concepts.
- (Q5) Tell us what you disliked about the extra-curricular competition to learn course concepts.

To generate annotations, we had one researcher inductively form themes for each question, assigning at most two themes to each non-trivial non-empty response. We received 328 such responses for (Q57), 113 for (Q30), and 85 for (Q5).

6.3.1 Thematic Validation and Interrater Reliability

In order to validate the original themes generated by our researcher, we employed interrater reliability with 2 other independent validators as reviewers. The validation set consisted of 20% of each question's thematically annotated responses chosen at random. The agreement between our generated themes and our first validator, measured by Jaccard Index scores, was $J = .15$ for (Q5), $J = .67$ for (Q30), and $J = .74$ for (Q57), showing strong agreement for (Q30) and (Q57). The researcher who generated the themes and the first validator met to discuss their disagreements regarding the themes for (Q5), then performed reannotation together. The revised themes were sent for revalidation with another independent validator. Agreement for the new annotations for (Q5) was $J = .81$. Statistics reported are based on the validated thematic labels by the original annotator.

6.3.2 Theme statistics

For (Q57), 259 (80% of 324) comments indicated that students did not have the time to commit to the competition and were too busy. 31 (9%) indicated the extra credit was too difficult to achieve, 20 (6%) indicated the base class assignment was already challenging, 18 (5%) indicated the reward for the competition was not worth the time and effort it would take to achieve, 11 (3%) indicated that they did not need the extra credit, and 11 (3%) indicated they were simply not interested. Other reasons represented at 1% or less were: lack of online support, unawareness of the competition, and adversity to competition in general.

For (Q30), 36 (32% of 113) participants indicated that the competition helped them further explore or apply course concepts. 29 (19%) indicated that they enjoyed the competitive and gamified aspect of the competition, 22 (19%) indicated that the competition made the assignment more interesting or fun, 20 (18%) indicated enjoying visibility of classmate performance, 13 (12%) indicated appreciating the integrated visualizations, and five (4%) indicated liking the optionality of the extra credit. Only one participant said they did the competition solely for the extra credit.

For (Q5), 26 (30% of 85) participants indicated they disliked something about the competition format or game design. 20 (24%) indicated that they did not feel they had the time or were too busy to compete well in the competition, 17 (20%) indicated a desire for better feedback systems for improvement, 12 (14%) indicated that competing or achieving extra credit was too difficult, six (7%) indicated adversity to competition, five (6%) indicated a wish to publicize winning strategies post-competition, and four (5%) indicated wanting more opportunities to interact with classmates in the competition. Only one participant indicated disliking the competition due to gamification.

6.4 Lessons Learned and Challenges

The development period for our competition extended over the course of five months, requiring constant iteration and improvement. The following principles were recurring themes in our process and are critical to building similar systems in the future.

6.4.1 Platform abstraction

We developed our platform around another game first, prior to integration with the class it was designed for. This strategy allowed our platform development to occur in parallel with course instructors' schedule for course development, rather than sequentially, which provided the greatest amount of flexibility to course instructors. Due to this initial abstraction of platform design away from course content, the only items that needed to change for a tight deployment timeline were the game visualization and the game script used on dedicated servers to simulate matches.

Enforcing abstraction also enhanced the re-usability of the platform: An undergraduate Artificial Intelligence course of 220 students in the Fall of 2025 has since re-used the platform for a mandatory project, demonstrating the platform's modularity. All that needed to change for the platform to host the game were the visualizations and the game-running script.

6.4.2 Class integration

When designing course-integrated tools, we found that maintaining constant communication with instructors was of the utmost importance, not only to coordinate development deadlines for the platform and course material but also to ensure compatibility between the platform and course content. An issue that arose during our time working with course instructors was the degree of separation between the student competitors and us. Due to the lack of a direct mode of communication between the students and us, we found it difficult to advertise to, announce to, and resolve issues with students promptly. This problem was further compounded by the fact that course instructors could not view or interact with competition data to maintain independence and prevent bias during instruction.

6.4.3 Optionality

Course instructors and developers came to the agreement that participation in the competition would not be required. Some participants in the competition listed this as a benefit due to experiencing a shift from external pressure to earn a grade to an intrinsic motivation for goal-oriented learning. However, had the entire class participated, statistical analysis for individual minority subgroups would have been more feasible than it was in our experience, because the number of participants would be an order of magnitude greater.

7 Limitations

This study has limitations that should be considered when interpreting the results. Our findings are based on one implementation within a single graduate course. Additionally, students who elected to take part in the competition may have been more motivated or engaged in course curriculum prior to participation. Our analysis also lacks pre-competition baseline measurements, making it difficult to establish causal relationships between the platform and learning outcomes.

8 Conclusion

In this paper, we detail our experience building a live, reusable, competitive adversarial game AI platform and integrating it into an existing college course at scale. The competition served as an optional, extra-credit portion for an assignment with features that provided timely, direct feedback to students, allowing them to develop solutions iteratively. Overall, students who responded to the optional competition post-survey tended to have on average positive perceptions of learning for the competition. We found that of the surveyed students, students who participated in the competition were on average two years older. There was also a significant correlation between the time surveyed participants spent on the competition and perceived learning benefit, indicating that for surveyed participants, more time spent correlated moderately positively with applying course content to solve problems and moderately positively with helping to develop confidence in the subject. Based on validated themes generated from survey responses, the primary barrier to participation for survey respondents in the competition was a lack of time. The platform was eventually reused for another undergraduate Artificial Intelligence course of 220 students as part of a mandatory assignment.

This work contributes to the growing body of research on automated educational technology in computer science classrooms. Specifically, we demonstrate the plausibility and potential learning benefits of hosting a competitive platform to teach core programming concepts, and detail the process and architecture via which it can be reproduced. We also describe the efficacy and reusability introduced by abstracting the platform from specific course content. In the future, we hope to leverage the reusability of our platform by integrating it into several other computer science courses, especially introductory programming language courses.

9 Acknowledgements

To Milap Naik and Paul Davis, course instructors for Graduate Introduction to Artificial Intelligence at the Georgia Institute of Technology, for their collaboration in implementing our competition as part of their curriculum.

References

[1] K. Siek and J. Battem, "Taulbee survey 2024: Annual report," 2025.

[2] C. T. Y. T. Stuart Zweben, Jodi L. Tims, "Acm-ndc study 2020-2021: Ninth annual study of non-doctoral granting departments in computing," *ACM Inroads*, vol. 12, Apr. 2021.

[3] M. Rõõm, P. Luik, and M. Lepp, "Learner success and the factors influencing it in computer programming mooc," *Education and Information Technologies*, vol. 28, no. 7, pp. 8645–8663, 2023.

[4] H. Burgiel, P. M. Sadler, and G. Sonnert, "The association of high school computer science content and pedagogy with students' success in college computer science," *ACM Trans. Comput. Educ.*, vol. 20, Apr. 2020.

[5] R. Vivian, K. Falkner, and N. Falkner, "Computer science students' causal attributions for successful and unsuccessful outcomes in programming assignments," in *Proceedings of the 13th Koli Calling International Conference on Computing Education Research*, Koli Calling '13, (New York, NY, USA), p. 125–134, Association for Computing Machinery, 2013.

[6] S. N. Liao, K. Shah, W. G. Griswold, and L. Porter, "A quantitative analysis of study habits among lower- and higher-performing students in cs1," in *Proceedings of the 26th ACM Conference on Innovation and Technology in Computer Science Education V. 1*, ITiCSE '21, (New York, NY, USA), p. 366–372, Association for Computing Machinery, 2021.

[7] B. C. Wilson, "A study of factors promoting success in computer science including gender differences," *Computer Science Education*, vol. 12, no. 1–2, pp. 141–164, 2002.

[8] N. Rountree, J. Rountree, A. Robins, and R. Hannah, "Interacting factors that predict success and failure in a cs1 course," *SIGCSE Bull.*, vol. 36, p. 101–104, June 2004.

[9] C. Stephenson, A. Derbenwick Miller, C. Alvarado, L. Barker, V. Barr, T. Camp, C. Frieze, C. Lewis, E. Cannon Mindell, L. Limbird, D. Richardson, M. Sahami, E. Villa, H. Walker, and S. Zweben, "Retention in computer science undergraduate programs in the u.s.: Data challenges and promising interventions," in *Proceedings of the ACM Conference*, (New York, NY, USA), Association for Computing Machinery, 2018.

[10] A. Kapoor and C. Gardner-McCune, "Understanding cs undergraduate students' professional identity through the lens of their professional development," in *Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '19, (New York, NY, USA), p. 9–15, Association for Computing Machinery, 2019.

[11] S. Hooshangi, R. Buxton, and M. Ellis, "Integration of practical computing skills and co-curricular activities in the curriculum," in *Proceedings of the 27th ACM Conference on on Innovation and Technology in Computer Science Education Vol. 1*, ITiCSE '22, (New York, NY, USA), p. 61–67, Association for Computing Machinery, 2022.

[12] M. Ortiz, K. Chiluiza, and M. Valcke, "Gamification in computer programming: Effects on learning, engagement, self-efficacy and intrinsic motivation," 10 2017.

[13] A. Ahmad, F. Zeshan, M. S. Khan, R. Marriam, A. Ali, and A. Samreen, "The impact of gamification on learning outcomes of computer science majors," *ACM Trans. Comput. Educ.*, vol. 20, Apr. 2020.

[14] T. L. Naps, G. Rößling, V. Almstrum, W. Dann, R. Fleischer, C. Hundhausen, A. Korhonen, L. Malmi, M. McNally, S. Rodger, and J. A. Velázquez-Iturbide, "Exploring the role of visualization and engagement in computer science education," in *Working Group Reports from ITiCSE on Innovation and Technology in Computer Science Education*, ITiCSE-WGR '02, (New York, NY, USA), p. 131–152, Association for Computing Machinery, 2002.

[15] J. C. Paiva, J. P. Leal, and A. Figueira, "Automated assessment in computer science education: A state-of-the-art review," *ACM Trans. Comput. Educ.*, vol. 22, June 2022.

[16] B. Ruth and J. R. Hott, "Auto-grading in computing education: Perceptions and use," in *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 1*, SIGCSETS 2025, (New York, NY, USA), p. 1008–1014, Association for Computing Machinery, 2025.

[17] J. DeNero, S. Sridhara, M. Pérez-Quiñones, A. Nayak, and B. Leong, "Beyond autograding: Advances in student feedback platforms," in *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '17, pp. 651–652, Association for Computing Machinery, Mar. 2017.

[18] S. Wasik, M. Antczak, J. Badura, A. Laskowski, and T. Sternal, "A survey on online judge systems and their applications," *ACM Comput. Surv.*, vol. 51, pp. 3:1–3:34, Jan. 2018.

[19] M. N. Giannakos, I. O. Pappas, L. Jaccheri, and D. G. Sampson, "Understanding student retention in computer science education: The role of environment, gains, barriers and usefulness," *Education and Information Technologies*, vol. 22, p. 2365–2382, Sept. 2017.

[20] J. Hattie and H. Timperley, "The power of feedback," *Review of Educational Research*, vol. 77, no. 1, pp. 81–112, 2007.

[21] S. Krusche, L. M. Reimer, N. von Frankenberg, and B. Bruegge, "An interactive learning method to engage students in modeling," in *2020 IEEE/ACM 42nd International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*, p. 12–22, Oct. 2020.

[22] J. A. Stone, G. Nagakalyani, S. Chaudhary, V. Apte, G. Ramakrishnan, and S. Tamilselvam, "Ta buddy: Ai-assisted grading tool for introductory programming assignments," in *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 2*, SIGCSE '25, (New York, NY, USA), ACM, 2025.

[23] J. W. Orr and N. Russell, "Automatic assessment of the design quality of python programs with personalized feedback," June 2021. arXiv:2106.01399 [cs].

[24] B. Marín, J. Frez, J. Cruz-Lemus, and M. Genero, "An empirical investigation on the benefits of gamification in programming courses," *ACM Trans. Comput. Educ.*, vol. 19, pp. 4:1–4:22, Nov. 2018.

[25] F. F.-H. Nah, Q. Zeng, V. R. Telaprolu, A. P. Ayyappa, and B. Eschenbrenner, "Gamification of education: A review of literature," in *HCI in Business* (F. F.-H. Nah, ed.), (Cham), p. 401–409, Springer International Publishing, 2014.

[26] M. Tasadduq, M. S. Khan, R. M. A. Nawab, M. H. Jamal, and M. T. Chaudhry, "Exploring the effects of gamification on students with rote learning background while learning computer programming," *Computer Applications in Engineering Education*, vol. 29, no. 6, p. 1871–1891, 2021.

[27] C. J. Hellín, F. Calles-Esteban, A. Valledor, J. Gómez, S. Otón-Tortosa, and A. Tayebi, "Enhancing student motivation and engagement through a gamified learning environment," *Sustainability*, vol. 15, p. 14119, Jan. 2023.

[28] S. Maharaj, N. Polson, and A. Turk, "Chess ai: Competing paradigms for machine intelligence," *Entropy*, vol. 24, p. 550, Apr. 2022. arXiv:2109.11602 [cs].

[29] D. Silver, A. Huang, C. Maddison, A. Guez, L. Sifre, G. Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, p. 484–489, Jan. 2016.

[30] S. Tao, A. Kumar, B. Doerschuk-Tiberi, I. Pan, A. Howard, and H. Su, "Lux ai season 3: Multi-agent meta learning at scale," Apr. 2024.

[31] C1 Games (formerly T1 Games), "TERMINAL." https://terminal.c1games.com.

[32] Battlesnake Developers, "Battlesnake official documentation." https://docs.battlesnake.com, 2024.

[33] MIT Battlecode, "Battlecode programming competition." https://battlecode.org/, 2025.

[34] M. Wollowski and J. Verkamp, "Effects of game tournaments on learning and classroom climate.," in *FLAIRS*, 2010.

[35] N. Lipovetzky and S. Sardina, "Pacman capture the flag in ai courses," *IEEE Transactions on Games*, vol. 11, no. 3, pp. 296–299, 2018.

[36] F. Chesani, A. Galassi, P. Mello, and G. Trisolini, "A game-based competition as instrument for teaching artificial intelligence," in *Conference of the Italian association for artificial intelligence*, pp. 72–84, Springer, 2017.

[37] J. Mao, "Autoscrim algorithm." https://github.com/battlecode/galaxy/blob/2c6ccf96e18f189f74a3b d5e4a9df07edaacd4b5/backend/siarnaq/api/teams/managers.pyL208, 2022. GitHub repository, accessed January 2025.

[38] P. G. Feijóo-García, C. Wrenn, A. Gomes de Siqueira, R. Ghosh, J. Stuart, H. Yao, and B. Lok, "Exploring the effects of user-agent and user-designer similarity in virtual human design to promote mental health intentions for college students," *ACM Trans. Appl. Percept.*, vol. 22, Nov. 2024.

[39] T. T. Diemer, E. Fernandez, and J. W. Streepey, "Student perceptions of classroom engagement and learning using ipads," *Journal of Teaching and Learning with Technology*, pp. 13–25, 2012.

[40] H. Wu and S.-O. L. and, "Can likert scales be treated as interval scales?—a simulation study," *Journal of Social Service Research*, vol. 43, no. 4, pp. 527–532, 2017.

[41] A. Joshi, S. Kale, S. Chandel, and D. Pal, "Likert scale: Explored and explained," *British Journal of Applied Science Technology*, vol. 7, pp. 396–403, 01 2015.