Large-scale Deployment and Evaluation of an Academic Integrity Module for Computing Students

Debarati Basu
basud@erau.edu
Embry-Riddle Aeronautical University
Daytona Beach, Florida, USA

Harini Ramaprasad
hramapra@charlotte.edu
University of North Carolina at Charlotte
Charlotte, North Carolina, USA

Landon Nalewaja
lnalewaj@charlotte.edu
University of North Carolina at Charlotte
Charlotte, North Carolina, USA

***Abstract*—This innovative practice full paper describes a large-scale implementation and evaluation of an academic integrity module tailored for students in the Computer Science Program at the University of North Carolina at Charlotte. The module aims to educate students about academic misconduct, its consequences, and strategies to prevent it.**

While various approaches exist to tackle academic misconduct in computing, our approach is unique in two ways. First, it is a *proactive* approach that seeks to educate and empower students on this important topic, to help them be *responsible* for their actions in the academic setting, and be *aware of* the potential consequences they may face if they resort to academic integrity violations, rather than face potentially punitive consequences. Second, we incorporate scenarios and information that are specifically relevant to the *Computing* discipline throughout the module.

This paper presents the module's design, outcomes from its large-scale deployment, student feedback on its effectiveness, and reasons students typically resort to academic integrity violations. Additionally, it discusses plans for continuous improvement based on student input and information about emerging tools and technologies (e.g., Artificial Intelligence powered tools). We expect that this paper will help guide the rapid adoption, deployment, and evaluation of such a program-specific academic integrity learning module at other institutions. We expect that our module itself can be adapted to other computing programs with only a small amount of effort.

I. Introduction

Academic integrity violations have been observed worldwide, affecting educational institutions of all sizes and types, from prestigious universities to large public ones, in both developed and developing nations [1]. Research indicates a rise in Academic Integrity (AInt) cases within Computer Science (CS) compared to other fields in recent year [2], [3]. Some of this may be attributed to easy and abundant access to CS materials and assistance online, along with the increased use of academic integrity violation detection tools. At our institution (UNC Charlotte) in USA, AInt violations in the CS program have surged from 12 cases in 2016-2017 to over 100 annually, despite reinforced academic conduct codes and explicit policies.

Many universities in the USA have honor codes to promote a culture of trust, responsibility, and ethical behavior among students. However, its prevalence and strictness vary across institutions. Further, its implementation and impact differ significantly in other cultural contexts due to varying educational values, enforcement mechanisms, and cultural attitudes toward cheating and academic integrity [4]. In addition to honor codes, there have been a variety of ways in which researchers have addressed academic misconduct in computing. For example, by developing computing-specific policies in addition to institution-level policies to address the unique nature of work in the discipline, by developing plagiarism detection tools for computing, and by incorporating pedagogical practices within a course that provide students with learning opportunities. An earlier study found that most strategies include educating students; empowering students; discouraging cheating (e.g., more awareness of potential consequences, making student work visible, etc.); reducing the benefits of cheating (e.g., low-stakes assessments); and making cheating difficult (e.g., individualizing assessments).

In contrast with approaches where instructors are solely responsible for stopping misconduct, educating and empowering students allows them to take responsibility for not cheating and helps them learn to behave with integrity. UNC Charlotte is an urban research institution that prioritizes an inclusive educational environment. More than 40% of students at UNC Charlotte come from underrepresented and first-generation backgrounds, including 25% in CS. With the goal of fostering a growth mindset [5], [6] and recognizing that students may unknowingly commit AInt violations, we adopt this strategy of *educating* them about misconduct and its consequences to curb such incidents [7].

To this end, we have developed a standalone, self-paced online AInt module tailored for CS students. Our approach is innovative in two ways. First, it is a *proactive* approach that seeks to educate students on this important topic, to help them be *responsible* for their actions in the academic setting, and be *aware of* the potential consequences they may face if they resort to academic integrity violations, rather than face potentially punitive consequences. Second, we incorporate scenarios and information that are specifically relevant to the *Computing* discipline throughout the module. To the best of our knowledge, no academic integrity-related module has been developed specifically for computing students.

By completing our module, students should be able to: a) recognize misconduct in academic scenarios, b) recognize the

consequences of academic misconduct, and c) recall strategies and resources to avoid academic misconduct. In Fall 2023, we deployed this module and an accompanying student survey in 35 undergraduate and graduate courses within the CS program at UNC Charlotte.

In this paper, we briefly present our module and the results of our large-scale deployment and evaluation, including student perceptions of the effectiveness of our module, and reasons why students may resort to academic integrity violations. We also present our strategy for continuous enhancement of our module to incorporate student feedback and information about new technologies (e.g., Artificial Intelligence tools). We expect that this will help guide the rapid adoption, deployment, and evaluation of such a program-specific academic integrity learning module at other institutions. We expect that our module itself can be adapted to other computing programs with only a small amount of effort.

The contributions of this study are:

- a large-scale deployment and evaluation of the effectiveness of our AInt module
- an analysis of students' perceptions on why they resort to academic integrity violations

The rest of the paper is organized as follows. Section II reviews relevant literature and Section III provides a brief overview of our module. Section IV presents details of our deployment and describes our data collection and analysis methods. Section V presents our analysis results, followed by a discussion of our results in Section VI. Section VII presents our conclusions and plans for future work.

II. Literature Review

A recent investigation into the evolution of academic integrity and computer ethics education within the CS realm over the past 50 years found that interest in this topic only spiked around 2017-2018 [8]. The field of Computer Science experiences a disproportionately higher rate of academic integrity violations compared to other disciplines [9]. It is not certain whether this is due to 1) truly higher rates of dishonesty; 2) the existence of more detection tools in the field; or 3) the nature of solutions in this field, especially to programming problems (i.e., there may only be a few efficient ways to solve a given problem) [9].

Riedesel et al. suggest that part of the issue stems from students' misunderstanding of their university's academic policies, noting that "a number of computing departments have created their own policies that extend, replace, or interpret their institutional policies in the context of computing education" [10]. In a study conducted by Stepp and Simon [11], 112 students were tested to evaluate their proficiency in generating scenarios exemplifying best practices in student collaboration. Through this experiment, the researchers aimed to unveil the students' cognitive processes and comprehension of various academic policies. It was observed that a significant portion of the scenarios devised by the students would violate their university's academic regulations, once again suggesting that students are not always aware of what constitutes academic misconduct, especially in the introductory programming courses.

The issue of academic misconduct in the CS discipline has been addressed in a variety of ways. One approach is the creation of CS-specific policies to address the unique nature of work in the discipline, which is often not covered by institution-level policies for student conduct. For example, a study proposed a model for developing and implementing an academic ethics policy (which encompasses academic integrity) that specifically addresses the challenges imposed by information technology, through evidence-based pedagogical practices in computer science and engineering [12]. Another approach is the development/enhancement of plagiarism detection tools for CS. For example, a study developed a tool that is able to detect academic misconduct in programs that only contain a few lines of code more accurately than traditional software plagiarism detection tools like MOSS and TMOSS [13].

In many cases, adding explicit and accessible course policies can significantly impact students' integrity levels within a course. For instance, in a study by Malan et al., the introduction of a 'regret clause' in the course syllabus enabled students to voluntarily disclose instances of academic dishonesty within a 72-hour window [14]. This approach prompted many students, including those not suspected of misconduct by faculty, to step forward and provide explanations for their violations. This technique provided a learning opportunity for students instead of penalizing them right away.

Sheard et al. conducted a nationwide investigation involving interviews with 30 computer science educators representing 25 universities. This study revealed a diverse array of tactics employed by educators to mitigate academic misconduct within introductory programming courses [7]. These strategies encompassed educational initiatives, empowerment measures, and deterrent methods against cheating, such as heightening awareness of potential consequences and increasing the visibility of student work. Additional strategies aimed to diminish the incentives for cheating, such as implementing low-stakes assessments, and to make cheating more challenging, such as customizing assessments for individual students.

Education in the context of Sheard et al.'s study [7] involved imparting knowledge about academic integrity principles and equipping students with pertinent resources and tools. Empowerment efforts centered on fostering supportive relationships with students and emphasizing learning over grades, thereby reducing the likelihood of cheating. The authors underscored the positive orientation of these strategies, emphasizing their role in cultivating an environment conducive to academic honesty. Furthermore, these strategies emphasize a shared responsibility between instructors and students, with instructors tasked with educating students about academic integrity while students bore the responsibility of adhering to ethical standards. Such an approach facilitates the development of integrity among students, in contrast to approaches solely reliant on instructor intervention to curb misconduct, although

the latter remains essential.

The emergence of generative artificial intelligence (AI) tools like ChatGPT, Copilot, and Microsoft's Bing provides students with access to resources that not only assist in problem-solving but also complete various computing tasks assigned in their curriculum. The opportunities and challenges of using ChatGPT in higher education are elaborated in [15], [16]. In a paper that documents the prompts from an author and responses by the ChatGPT chatbot on Plagiarism, the chatbot suggests that students need to understand the implications of using ChatGPT and avoid its misuse [17]. It recommends the use of stricter policies and regulations and the education of students on how these technologies should be used ethically. A recent study by Joshi et al. [18] thoroughly examined ChatGPT's capabilities in tackling common Computer Science queries. They tested ChatGPT with questions spanning topics from applied computing to data structure search algorithms. The evaluation highlighted the importance of providing clear and concise context in prompts to ensure consistent and accurate outputs from ChatGPT. The study concluded that relying on ChatGPT for coding problems may inadvertently lead students to produce incorrect answers, posing a risk of "self-sabotage."

Overall, our review of the literature and our experiences in this area have shown us that it is important to help students understand and be aware of academic integrity violations, and ethical use of various resources and technologies. In a study, Jiménez et al. [19] developed an academic integrity module to be used specifically by engineering students. This module was part of an "ethics across the engineering curriculum program as a space for students and professors to connect engineering and its ethical implications." They conclude that the module impacted students in two ways: students' learning processes and their ethical awareness/evaluations. We achieve a similar goal through an online module on academic integrity targeted at *computing* students. Although there have been several strategies found in the literature to address academic misconduct, we have not found any academic integrity module developed targeting students in a computing discipline, where the academic integrity cases are comparatively higher.

III. Academic Integrity (AInt) Module

Our AInt module is specifically targeted towards CS students. Our overall goal is to increase student awareness about academic misconduct and educate them on strategies and resources that they can use to avoid academic misconduct. The guidelines and process for our AInt module design are explained in an earlier paper [20]. Here we briefly describe the objectives and content of the module. After completing our module, students should be able to:

- recognize misconduct in academic scenarios;
- recognize the consequences of academic misconduct;
- recall strategies and resources to avoid academic misconduct.

To meet the first learning objective, our module introduces and defines the different types of academic misconduct identified by our institution, including Cheating, Falsification, Plagiarism, etc. For each type of misconduct, the module presents several examples tailored to CS students. To gauge students' understanding, our module includes a) a quiz featuring CS-specific academic scenarios, prompting students to determine whether they constitute violations of academic integrity; and b) a quiz comprising anonymized case studies drawn from real-life situations encountered by faculty within our CS program, tasking students with identifying acceptable student responses from a provided list of options.

Our module's second learning objective aims to cultivate students' awareness of the repercussions associated with academic misconduct. It presents important reasons for abstaining from such behaviors, including how academic misconduct can impede students' educational progress, diminish the efforts of others, and lead to potential repercussions both within their academic journey and in their future professional endeavors. The module also outlines the array of sanctions established by our institution for various types of academic misconduct. Penalties may include a lowered course grade, mandatory resubmission of academic assignments, academic suspension, etc., in addition to a formal record documenting the violations.

Our module's third objective is to help students remember strategies and resources to avoid academic misconduct. To meet this goal, we offer practical tips on avoiding such misconduct. These include understanding our institution's policies on academic integrity, knowing what is allowed or disallowed within specific courses, how to use and cite online sources correctly, documenting code, seeking help when needed, etc. We also share strategies for exams, paper writing, and collaboration. Furthermore, we provide links to various resources for students to succeed, both within our institution, like the writing center, library, and academic support center, and outside resources covering note-taking, time management, and study skills. Lastly, we provide contact details for the office overseeing academic integrity and student conduct issues at our institution.

As Generative AI tools become more prevalent in teaching and learning, the conversation around their use and its impact on academic integrity is becoming increasingly critical. In our module, we provide examples of cheating using AI tools. We also outline types of course policies that instructors may have related to the use of AI tools and encourage students to carefully review those policies, as well as seek clarification from their instructors when necessary. In addition, students are asked to interpret acceptable actions within scenarios related to AI tool use within the module quizzes, demonstrating their aptitude in the recognition of academic misconduct. In future work, we plan to incorporate a more comprehensive discussion on the use of AI tools within our module.

At the end of our module, we include two more elements: a quiz to assess students' grasp of the module's main points and objectives, and a survey with questions that encourage students to reflect on their learning and experiences with the module. The survey is detailed in the next section.

IV. METHOD

We utilized a mixed-method research design to conduct this Institutional Review Board (IRB) approved study. In this paper, we will present the results of the following research questions:

- How effective was the AInt module in helping CS students gain knowledge about academic misconduct?
- Why do students resort to academic integrity violations?

In the remainder of this section, we present details about the deployment of the module, data collection, and analysis.

A. Study Deployment

In Fall 2023, our AInt module was deployed in 35 computing courses, and a Teaching Assistant Training Course within the CS program at UNC Charlotte. These computing courses are listed below and categorized by lower-level and higher-level undergraduate courses and graduate-level courses. This highlights the module's applicability across various types of CS courses. Instructors had the flexibility to integrate the module at any time during the semester. As students progressed within the CS program, they may have encountered this module multiple times. This was a deliberate effort to stress the importance of academic integrity and reinforce students' learning and understanding of academic misconduct in the CS field. Furthermore, this way, they are exposed to any updated policies and information we keep incorporating into the module. Among these students, **486** students consented to participate in this study.

Lower Level Courses: These are freshman and sophomore-level courses that are required for all computing students for their CS undergraduate degree.

- Introduction to Computer Science I
- Introduction to Computer Science II
- Computing Professionals
- Computer Science Program, Identity, Career
- Topics in Computer Science
- Introduction to Computer Systems
- Data Structures and Algorithms
- Logic and Algorithms
- C Programming

Upper-Level Courses: These are junior and senior-level courses that are taken by computing students in their area of concentration or as an elective for their CS undergraduate degree.

- Computers and Their Impact on Society
- Undergraduate Research
- Intro to Artificial Intelligence
- Database Design and Implementation
- Intro to Computer Networks
- Computing Leaders Seminar
- Programming Languages
- Visual Analytics
- Intro to Game Design and Development
- Interaction Design Projects

Graduate Level Courses: These courses are taken by Masters and PhD students in the computing program to complete their CS graduate degree.

- Survey of Programming Languages
- Visual Analytics
- Intro to Game Design and Development
- Software System Design & Implementation
- Illustrative Visualization
- Computer Communication & Networks
- Database Systems
- Interaction Design Projects
- Principles of Information Security and Privacy
- Data Privacy
- Applied Machine Learning

B. Data Collection

We designed an electronic reflection survey that we asked students to complete right after they completed the AInt module. The survey had the following four parts. In this paper, we utilize parts one and four to answer the first research question, and part three to answer the second research question.

- Part 1: To find the effectiveness of our AInt module, we used 5-point Likert scale questions on students' agreement with the usefulness of the module and whether it increased their knowledge / awareness about academic integrity. Also, we listed each part of the module and asked students to choose the parts that they found to be most helpful.
- Part 2: To assess the objectives of the module, we asked 5-point Likert scale questions to find students' agreement with the objectives of the module, and how likely they will be more careful to avoid academic integrity violations after having completed this module
- Part 3: To find reasons why students resort to academic integrity violations, we asked them an open-ended question.
- Part 4: We asked an open-ended question to collect students' additional feedback on the module, in an effort to further improve the module.

C. Data Analysis

To answer the first research question, there were three quantitative questions (Part 1 as indicated above). We used descriptive statistics to analyze the quantitative data to show how useful the module was to the students, whether the module increased their knowledge / awareness about academic integrity, and which parts of this module were most helpful to them. We coded the open-ended responses (Part 4 as indicated above), to summarize the feedback given by students to improve the module.

The open-ended question asking students to state reasons why they believe students may resort to academic integrity violations (Part 3 as indicated above) generates qualitative data. To analyze this data, we employed a three-step thematic analysis approach [21]. Thematic analysis is appropriate because we sought to understand the range of reasons students could resort

to academic violations [22]. First, we went through all student responses and created codes to capture the different reasons mentioned in them. These codes were carefully crafted to accurately represent the diverse range of reasons provided by the students. Second, we discussed the resulting codes, looked for patterns among the codes, and combined related codes into categories. We reviewed the categories and compared them once again with the student responses to assess whether they portrayed an accurate representation of student's opinions. Finally, we grouped related categories into broader themes that encapsulate the collective meaning and significance of the grouped categories.

V. Results

A. Overall effectiveness of the module

We now present the results to demonstrate the effectiveness of our AInt module. In the survey, when we asked students about the usefulness of the module, 65.88% (481) of the student participants mentioned that it was extremely or very useful to them, and it was moderately useful to an additional 24.24% (177) of the participants. Further, 86.3% (630) of the participants agreed or strongly agreed that completing this module increased their knowledge / awareness about academic integrity. Table I shows the different parts of the module and the corresponding number and percentage of participants, in descending order, who found those parts most useful. Overall, it seems each part of the module was useful to at least 36% of the participants. Most participants (66%) found *definitions of different types of academic integrity violations* to be the most useful. More than 50% of the participants found *examples of academic integrity violations for each type including the CS-specific examples*, and the *ability to determine whether a scenario is an academic integrity violation or not* to be the most useful. About 40 to 50% of the participants thought that *the strategies to avoid academic misconduct (e.g., documenting your code)*, *interpreting acceptable student actions in a given scenario*, and *resources for student success (e.g., writing resource center, time management development)* to be most useful. These parts were followed by *the reasons to avoid academic misconduct (e.g., hampers learning*, and *academic misconduct sanctions (e.g., suspension).*

TABLE I
Module parts and number and percentage of students it was useful for

Module Parts	# Responses (% Responses)
Definitions of different types of academic integrity	481 (66%)
Examples of academic integrity violations for each type	424 (58.16%)
Examples of academic integrity violations for each type (CS-specific examples)	417 (57.2%)
Determining whether a scenario is an academic integrity violation or not	371 (50.9%)
Strategies to avoid academic misconduct (e.g., documenting your code)	343 (47.05%)
Interpreting acceptable student actions in a given scenario	309 (42.38%)
Resources for student success (e.g., writing resource center, time management development)	300 (41.15%)
Reasons to avoid academic misconduct (e.g., hampers learning)	270 (37.03)
Academic misconduct sanctions (e.g., suspension)	264 (36.21%)

B. Students' perceptions on why they resort to violating academic integrity

In our student reflection survey, we asked students the following open-ended question: "*In general, why do you think students might resort to violating academic integrity?*" Among the **729** students who consented to participate in our study, **568** of them responded to this open-ended question. We analyzed the qualitative data generated from these student responses using the 3-step thematic analysis approach outlined in Section IV. In the first step, we derived a total of **605 unique codes** from the responses — note that a single student's response sometimes contained multiple reasons and hence contributed to multiple codes. In the second step, we grouped the 605 codes into **14 different categories**. In the third step, **four themes** emerged from the 14 categories.

Table II shows the results of the second and third steps of our thematic analysis. The table is grouped by the four themes. Within each group, the first column shows the categories within the corresponding theme and the second column shows the number and percentage of unique codes that contributed to each category. In the third column, we provide a sample student response relevant to that category.

a) Student Emotions: This theme comprises four categories reflecting different kinds of student emotions: the pursuit of high grades or GPA maintenance, fear of failure, heightened stress, pressure, or mental health challenges, and a general sense of being overwhelmed. A total of 247 student codes (approximately 38%) fell within this overarching theme.

b) Student Productivity Barriers: This theme covers five categories outlining obstacles to student productivity, potentially leading to academic integrity violations. These include issues like ineffective time management, procrastination, or inadequate planning; tendencies towards laziness, weak work ethics, and opting for convenience in completing assignments; reluctance to complete assignments, apathy, lack of motivation/drive, and indifference towards the course; insufficient time dedicated to learning, ineffective study methods; and embarrassment/shyness in seeking assistance. A total of 297 codes (approximately 47%) fell into this theme.

c) Lack of Understanding of Consequences: This theme encompasses two categories highlighting students' lack of comprehension regarding academic integrity policies, unawareness of consequences, and a belief in escaping detection

or avoiding penalties. This theme was reflected in 22 codes, accounting for less than 4% of the total.

d) Course Related Issues: The final theme encapsulates three categories addressing course-related challenges, such as unclear assignment formats or difficulty comprehending instructions; complexity or lack of understanding of course content; tight deadlines, or heavy workloads; and insufficient teacher support, resources, or instruction time. This theme was reflected in 39 codes, constituting approximately 6% of the total.

VI. Discussion

Based on our data analysis from this large-scale implementation, we found that our AInt module was useful to most students, and more than 86% of the students thought this module increased their knowledge/ awareness about academic integrity, with the definitions, examples, and CS-specific examples being the three most useful parts of the module. This result is consistent with our findings from a pilot deployment of our module as well [20]. We believe these definitions and examples effectively helped students clearly distinguish and identify academic integrity scenarios relevant to their context. It addresses the concern found in [11] that students often perceive certain academic misconduct scenarios as acceptable scenarios. Further, we found that even though some parts of the module were identified as being most useful by more students than others, *every* part of the module was identified as being most useful by at least 36% of the participants. These results demonstrate that the module with all its content is effective in educating students about academic misconduct.

One significant challenge in assessing our module's effectiveness lies in the absence of baseline data regarding the frequency of academic integrity violations in the specific courses where our module has been implemented. While UNC Charlotte has recently started collecting data on academic integrity violations at the course level, FERPA and student privacy regulations restrict their ability to provide detailed data to researchers. Additionally, anecdotal evidence suggests that faculty members may not consistently report all academic integrity violations due to the time-intensive nature of the reporting process, which means that reported figures may not fully represent the actual number of incidents. Furthermore, even with accurate data, attributing any observed decrease in academic integrity violations directly to our module remains challenging, as various other factors could also influence the occurrence of such violations. Also, we need to remember that students from the different courses voluntarily participated in the study, and findings may be influenced by their characteristics [23].

Additionally, in this study, utilizing thematic analysis of the students' open-ended responses, we identified various reasons students may resort to academic integrity violations. The most popular are students' productivity barriers, such as time management, and poor work ethic. This is followed by the reasons that relate to different kinds of student emotions, such as stress and fear of failure. A comparatively smaller number of students highlighted reasons that relate to a lack of understanding of the consequences of academic integrity violations, and course-related issues such as confusion with course content and high workload.

Understanding these reasons can greatly benefit faculty members in various ways. Firstly, it provides insights into the underlying issues affecting student behavior, enabling instructors to address some of these issues. For example, by recognizing productivity barriers, faculty can offer guidance on time management techniques and cultivate a stronger work ethic among students. Likewise, acknowledging the students' emotions allows instructors to implement supportive measures in the classroom. Furthermore, awareness of students' misunderstandings regarding academic integrity consequences and course-related challenges enables faculty to enhance educational resources and clarify expectations in the course. Thus, we hope that these results will equip faculty with valuable tools to better support and guide students toward academic integrity and success. Additionally, these reasons guide us in developing strategies to avoid academic misconduct that we will integrate into our module.

Academic integrity violations present a continuous challenge with new things to learn and address. There will always be new scenarios and information to add to our academic integrity module. For example, as the number of AI tools continues to grow, we can expect a variety of new academic misconduct scenarios to emerge in the future. It is crucial that we equip students with the necessary guidance to navigate these scenarios and avoid potential misconduct. For this, we engage in continuous improvement of our module every year during the summer months where we incorporate students' feedback and information related to new technologies. For example, the participants of our study gave invaluable feedback on what needs to be added to the module such as: removing repetitive material, providing more information on proper AI usage, requiring first and second-year students to complete the module, providing fewer words and more videos, and many more pieces of information. Before the deployment of our module in the next academic year, we will strive to focus on the areas that receive the most attention and are guaranteed to improve the students' awareness of academic misconduct.

VII. Conclusion

In this paper, we present details of a large-scale deployment and evaluation of the effectiveness of an academic integrity (AInt) module that we developed in our prior work, and an analysis of students' perceptions of why they resort to academic integrity violations. Our results from deployment within 35 courses show that the module was effective in increasing students' awareness about academic integrity violations. Additionally, we found several factors that contribute to students' academic misconduct including student emotions, productivity barriers, lack of understanding of consequences, and course-related issues. Student feedback on the module is also summarized in this paper, which will be incorporated during our next yearly update of the module. We expect that

TABLE II
THEMES AND CATEGORIES DERIVED FROM STUDENT RESPONSES.

Categories	# Responses (% Responses)	Example Student Quotes
Theme 1: Student Emotions		
Desire for Good Grades	50 (8.26%)	"College is hard. Life is hard. Sometimes you just want to get a good grade and move on."
Fear of Failure	54 (8.92%)	"Students might violate academic integrity due to pressure to succeed, lack of time, or fear of failure."
Stress / Pressure	113 (18.68%)	"Stress can be a part of this violation."
Desperation	30 (4.96%)	"Students might resort to violating academic integrity because they don't have any more options."
Theme 2: Student Productivity Barriers		
Time Management	144 (23.80%)	"Most likely because of a lack of time/poor time management and having too many responsibilities."
Poor Work Ethic	25 (4.13%)	"Lethargy, lack of time management, poor work ethic, etc."
Lack of Motivation	106 (17.52%)	"Many students do not want to put the effort required into actually learning so they resort to cheating."
Poor Study Methods	15 (2.48%)	"poor time management, study methods, lack of understanding/commitment"
Reluctant to get Help	7 (1.16%)	"I think it may have something to do with either embarrassment/shyness to reach out for help on an assignment..."
Theme 3: Lack of Understanding of Consequences		
Does not Understand Policies / Consequences	5 (0.83%)	"Students may resort to violating academic integrity due to a perceived lack of consequences, jeopardizing their learning and ethical development"
Assumption of Getting Away	17 (2.81%)	"Because they feel they can get away with it if they do it properly"
Theme 4: Course Related Issues		
Confusion with Course Content	7 (1.16%)	"...they do not fully understand concept needed to complete any assignment or exams in order to succeed"
High Workload	5 (0.83%)	"Overwhelming academic workloads and tight deadlines "may push students to resort to cheating."
Lack of Course Support / Resources	27 (4.46%)	"They don't know of available resources and are afraid of falling behind their peers."

our AInt module, its deployment, and findings will help in the development and evaluation of similar program-specific AInt learning modules that can mitigate AInt violations. If any instructor / practitioner would like to adapt this specific module, they can contact the authors of this paper.

In the future, besides improving the content of the module according to student and faculty feedback, we plan to develop a generic version of the module with instructions on how it can be adapted to a specific institution's computing program. Furthermore, we plan to develop a version of this module targeted at CS faculty members and teaching assistants, since it is crucial that they are also aware of academic misconduct.

***Index Terms*—Academic integrity, CS education, evaluation**

ACKNOWLEDGMENT

This material is based upon work supported by the College of Computing and Informatics at UNC-Charlotte, and Engineering Fundamentals Department at Embry-Riddle Aeronautical University.

REFERENCES

[1] K. Ahmed, "Academic integrity: Challenges and strategies for asia and the middle east," *Accountability in research*, vol. 27, no. 5, pp. 256–270, 2020.

[2] W. H. Murray, "Cheating in computer science," *Ubiquity*, vol. 2010, no. October, 2010.

[3] L. M. Krieger, "Stanford finds cheating—especially among computer science students—on the rise," *San Jose Mercury News, June*, vol. 8, p. 2010, 2010.

[4] N. T. Hendy, N. Montargot, and A. Papadimitriou, "Cultural differences in academic dishonesty: A social learning perspective," *Journal of Academic Ethics*, vol. 19, no. 1, pp. 49–70, 2021.

[5] C. Dweck, "What having a "growth mindset" actually means," *Harvard Business Review*, vol. 13, no. 2, pp. 2–5, 2016.

[6] K. Quille and S. Bergin, "Promoting a growth mindset in cs1: Does one size fit all? a pilot study," in *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education*, 2020, pp. 12–18.

[7] J. Sheard, M. Butler, K. Falkner, M. Morgan, and A. Weerasinghe, "Strategies for maintaining academic integrity in first-year computing courses," in *Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education*, 2017, pp. 244–249.

[8] N. Najjar, M. L. Maher, and M. Mohseni, "A 50-year retrospective on academic integrity and computer ethics in cs education," in *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education*, 2021, pp. 1283–1283.

[9] R. Fraser, "Collaboration, collusion and plagiarism in computer science coursework," *Informatics in Education*, vol. 13, no. 2, pp. 179–195, 2014.

[10] C. P. Riedesel, A. L. Clear, G. W. Cross, J. M. Hughes, Simon, and H. M. Walker, "Academic integrity policies in a computing education context," in *Proceedings of the Final Reports on Innovation and Technology in Computer Science Education 2012 Working Groups*, ser. ITiCSE-WGR '12. New York, NY, USA: Association for Computing Machinery, 2012, p. 1–15. [Online]. Available: https://doi.org/10.1145/2426636.2426638

[11] M. Stepp and B. Simon, "Introductory computing students' conceptions of illegal student-student collaboration," in *Proceedings of the 41st ACM technical symposium on Computer science education*, 2010, pp. 295–299.

[12] C. P. Riedesel, E. D. Manley, S. Poser, and J. S. Deogun, "A model academic ethics and integrity policy for computer science departments," *ACM SIGCSE Bulletin*, vol. 41, no. 1, pp. 357–361, 2009.

[13] A. Kassil and M. Ball, "Active academic integrity." in *SIGCSE*, 2021, p. 1284.

[14] D. J. Malan, B. Yu, and D. Lloyd, "Teaching academic honesty in cs50," in *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, 2020, pp. 282–288.

[15] D. R. Cotton, P. A. Cotton, and J. R. Shipway, "Chatting and cheating: Ensuring academic integrity in the era of chatgpt," *Innovations in education and teaching international*, vol. 61, no. 2, pp. 228–239, 2024.

[16] M. Khalil and E. Er, "Will chatgpt g et you caught? rethinking of plagiarism detection," in *International Conference on Human-Computer Interaction*. Springer, 2023, pp. 475–487.

[17] M. R. King and ChatGPT, "A conversation on artificial intelligence, chatbots, and plagiarism in higher education," *Cellular and molecular bioengineering*, vol. 16, no. 1, pp. 1–2, 2023.

[18] I. Joshi, R. Budhiraja, H. Dev, J. Kadia, M. O. Ataullah, S. Mitra, H. D. Akolekar, and D. Kumar, "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*, ser. SIGCSE 2024. New York, NY, USA: Association for Computing Machinery, 2024, p. 625–631. [Online]. Available: https://doi.org/10.1145/3626252.3630803

[19] L. O. Jiménez, E. O'Neill-Carrillo, and M. Rodríguez, "An introductory learning module on ethics and academic integrity for freshman engineering students," in *2009 39th IEEE Frontiers in Education Conference*, 2009, pp. 1–6.

[20] D. Basu and H. Ramaprasad, "Design and evaluation of an academic integrity module for computer science students," in *2023 ASEE Annual Conference & Exposition*, 2023.

[21] J. Green, K. Willis, E. Hughes, R. Small, N. Welch, L. Gibbs, and J. Daly, "Generating best evidence from qualitative research: the role of data analysis," *Australian and New Zealand journal of public health*, vol. 31, no. 6, pp. 545–550, 2007.

[22] M. E. Kiger and L. Varpio, "Thematic analysis of qualitative data: Amee guide no. 131," *Medical teacher*, vol. 42, no. 8, pp. 846–854, 2020.

[23] J. Arias, J. Swinton, and K. Anderson, "Online vs. face-to-face: A comparison of student outcomes with random assignment." *E-Journal of Business Education and Scholarship of Teaching*, vol. 12, no. 2, pp. 1–23, 2018.