

Standards-Based Grading in Undergraduate Courses for Technology Majors

Ruth Lamprecht ✉ 

Department of Mathematics and Computer Science, Mount St. Mary's University,
Emmitsburg, MD, USA

Jonathan McCurdy ✉ 

Department of Mathematics and Computer Science, Mount St. Mary's University,
Emmitsburg, MD, USA

Melanie Butler ✉ 

Department of Mathematics and Computer Science, Mount St. Mary's University,
Emmitsburg, MD, USA

Brian Heinold ✉ 

Department of Mathematics and Computer Science, Mount St. Mary's University,
Emmitsburg, MD, USA

Daniel Salinas Duron ✉ 

Department of Mathematics and Computer Science, Mount St. Mary's University,
Emmitsburg, MD, USA

Abstract

This paper outlines the methods employed by several instructors within a single department to implement standards-based assessments. The authors began integrating standards across multiple courses in their computer science, cybersecurity, data science, and mathematics programs. This shift was driven by a desire to promote equity in grading and to address the growing influence of artificial intelligence, which can obscure a student's true understanding. In this work, the authors examine the supporting research that guided their motivation and informed their implementation of various grading techniques. With an emphasis on courses involving technology, they also detail the processes they use to manage the new assessments, provide examples of assessment questions, and share key lessons learned in making this transition successful for both instructors and students. This work addresses a significant gap in the literature, as there appears to be a notable lack of resources on the application of standards-based grading in technical disciplines.

2012 ACM Subject Classification Social and professional topics → Computing education; Social and professional topics → Student assessment

Keywords and phrases Alternative Grading, Standards-Based Grading, Computer Science

Digital Object Identifier 10.4230/OASISs.ICPEC.2025.10

Supplementary Material *Collection:* https://figshare.com/collections/Standards-based_Grading_in_Technology_Courses/7900184

1 Introduction

To better support students and enhance learning outcomes, the authors began exploring various alternative grading methods at their small private liberal arts university. Their goal was to identify strategies that would allow students to demonstrate mastery of course content while improving student retention – all while maintaining academic rigor and promoting fairness and accuracy in grading. Alternative grading offers several mechanisms to achieve these aims, including decoupling learning from grading, encouraging students to engage with instructor feedback, promoting learning through mistakes, and providing opportunities for reassessment without penalty.



© Ruth Lamprecht, Jonathan McCurdy, Melanie Butler, Brian Heinold, and Daniel Salinas Duron;
licensed under Creative Commons License CC-BY 4.0

6th International Computer Programming Education Conference (ICPEC 2025).

Editors: Ricardo Queirós, Mário Pinto, Filipe Portela, and Alberto Simões; Article No. 10; pp. 10:1–10:14

OpenAccess Series in Informatics



OASIS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

10:2 SBG in Technology Courses

This paper aims to have broad applicability stemming from the diversity of the author's approaches. Over the past two years, the authors independently transitioned from traditional grading systems to a standards-based approach. Standards Based Grading (SBG) emphasizes small, re-takeable formative assessments aligned with specific learning objectives, as opposed to large, summative evaluations. The transitions varied among instructors: some fully adopted the new model immediately, while others introduced it gradually. In some cases, the revised courses were new preparations for the instructors, whereas in others, the same courses had previously been taught using traditional grading methods. The range of courses included skills-based introductory computer science classes, several "information-based" courses in computer science and cybersecurity, theoretical upper-level computer science courses, and hands-on data science courses.

Traditionally, most departmental courses assessed students with approximately 50% of the grade based on "out-of-classroom" work (such as homework and projects), and the remaining portion based on "in-classroom" assessments (such as quizzes and exams). However, with the emergence of Large Language Models (LLMs) and Generative AI tools like ChatGPT, the authors began to reconsider the emphasis placed on out-of-classroom assessments. Recognizing that these tools are easily accessible and may influence how students engage with assignments, the authors sought to ensure that students were developing a deep understanding of the material. This led to a shift in grading weight toward in-class assessments, where instructors have greater control over the evaluation environment. Standards-Based Grading (SBG) offered an effective solution, enabling the use of frequent, low-stakes, in-class assessments while moving away from high-stakes, one-time exams.

Although the weight of out-of-classroom work was reduced, homework and projects remained integral to the curriculum. These assignments continued to support learning outside the classroom and allowed for the assessment of more complex or integrated skills that do not fit neatly into a single learning objective. In addition, homework provided students with early exposure to the types of questions used in standards-based assessments, incentivizing engagement with the course material. Importantly, the authors emphasize that standards-based grading does not need to be applied uniformly across all components of a course. Certain aspects such as exams or skill-based assessments can follow the SBG model, while others, like projects or homework, may retain traditional grading structures or even a different alternative grading approach. This hybrid approach allows instructors to tailor assessment strategies to the goals of the course while also allowing them to experiment with ways to better support student learning.

2 Background

Mastery-Based Grading is an assessment system that provides students with a clear set of objectives and evaluates them based on their eventual mastery of the material [5]. Unlike traditional grading models, Mastery-Based Grading allows students to submit work, receive feedback, revise or retry the assignment, and resubmit it without penalty. A specific implementation of this approach is Standards-Based Grading (SBG), which typically involves low-stakes formative assessments aligned with clearly defined course objectives [19, 11]. Rather than focusing on earning a particular percentage, students aim to demonstrate mastery of specific learning goals. As a result, assignments are often evaluated on a "pass/fail" basis or using a standards-based rubric that emphasizes levels of mastery (e.g., emerging, proficient, exemplary) instead of traditional point values, with opportunities for revision and resubmission until mastery is achieved.

A key feature of SBG is its shift away from infrequent, high-stakes assessments, such as major exams that offer only a single chance to demonstrate learning, toward more frequent, lower-stakes evaluations. These ongoing assessments provide students with multiple opportunities to show understanding over time. Consequently, this model is often associated with reductions in student stress and test anxiety [14, 17, 12, 4]. Furthermore, the ability to revise and reattempt assessments encourages a growth mindset [2], as students are empowered to view intelligence as something that can be developed through effort and persistence.

Student reflection on feedback is another cornerstone of the SBG approach. In traditional grading systems, students often glance at their score and overlook the instructor's feedback, missing valuable opportunities for learning and improvement. When the emphasis is solely on the grade, feedback is frequently ignored, which hinders growth [13]. In contrast, SBG requires students seeking to improve their performance to actively engage with instructor feedback. This reflection is necessary before they revise or resubmit their work, reinforcing the learning process.

Although numerous studies have explored the implementation of SBG in mathematics [5], relatively few have examined its use in computer science education. One such study by McCane et. al. [15] compared students in a Java course who had previously taken either a mastery-based or traditional Python course. The findings indicated that students from the mastery-based course performed better, particularly those who had previously struggled. Notably, high-performing students were unaffected by the grading model. However, a longitudinal follow-up by the same authors [16] found that the absence of deadlines for mastery-based assessments led to increased procrastination. Additional research supports the claim that mastery- and standards-based grading better supports lower- and middle-performing students in computer science contexts [18].

The CS Equitable Grading Practices group has also contributed to the field by creating a community of practice that shares resources relevant to a variety of grading models, including specifications grading, which shares similarities with SBG [7]. Moreover, a study by Fine [9] that explored the integration of competency-based grading with equity-focused strategies and Universal Design for Learning found that this hybrid model fostered increased student engagement and deeper understanding through more frequent interactions.

Recent scholarship also underscores the need to revisit assessment practices in light of the growing influence of AI tools. Educators are increasingly concerned with issues related to identity verification, plagiarism, and the assurance of authentic learning [10]. They must remain vigilant in upholding the core objectives and principles of education [6]. One study concluded that "Assessment was seen as needing to shift towards more in-person and invigilated tasks that involved greater authenticity, personalisation, higher-order thinking, and disclosure of sources" [3]. This affirmation of the need for reform has further validated the authors' ongoing application and study of standards-based grading.

3 Courses

In general, the instructors implemented SBG in courses where learning objectives could be clearly articulated and broken down into specific, measurable standards. The instructors who adopted this approach teach a diverse range of courses across the department, including mathematics, computer science, cybersecurity, and data science. This paper primarily focuses on the computer science and data science courses in which SBG has been applied. The courses and standards included for this work are:

10:4 SBG in Technology Courses

Lower-Division Courses (1st and 2nd Year):

- Introduction to Computer Science I and II (Python and Java respectively)
- Data Structures and Algorithms
- Principles of Software Development
- Introduction to Data Science
- Machine Learning

Upper-Division Courses (3rd and 4th Year):

- Computer Security
- Operating Systems
- Database Management Systems
- Algorithms
- Theory of Computation

Introductory computer science courses typically run five sections per year, each enrolling 20–25 students. Since different instructors teach these courses each semester, maintaining consistency in course content is essential for ensuring students' continued success in subsequent classes. Creating a shared set of standards that all instructors can use – while still allowing flexibility in how they implement them – strikes a balance between academic freedom and curricular consistency. This approach offers a clear record of the material covered, which is valuable for instructors teaching later courses in the sequence.

A small set of examples of the standards and some assessment questions used for the courses in this paper can be found in Appendices A and B. The authors have created a repository of the standards from courses they have taught that can be found online at figshare – SBG in Technology Courses.

4 Methods of Assessment

An important avenue of evaluating student understanding and mastery in a standards-based course is with low-stakes re-takeable assessments, which are referred to as standards. The design of the standards aims to target one specific learning objective or a lesson. The format varies according to the material. Within the department they have ranged from interpreting code/results, having the student program by hand, demonstrating understanding of theoretical concepts, or coding a solution up on the computer. These standards were usually based on the material covered in the previous week's lecture and homework assignments, allowing students time to interact with the content, practice, and then be assessed on the material.

The implementation and set-up of alternative grading can be tailored to suit the preferences and style of each instructor. In the following sections, the authors discuss different approaches to administering, grading, and weighting standards in the final grade, allowing instructors to combine various methods to create the most effective system for them. Different combinations of these approaches have been used in all of the classes taught by the authors.

4.1 Administering Standards

One decision point for incorporating SBG into a course is when to assess the standards. The authors found a few options that fit well in their courses without a loss of class time for learning.

Administering Standards Approach 1 (Regularly in class). With this approach, students encounter anywhere from 1 to 3 new standards each week, typically based on the previous week's lecture. Depending on how often the class meets they can be done during the last 15-20 minutes of class on a single day. This regular assessment schedule helps students stay engaged with the material and provides frequent opportunities to demonstrate mastery. It also allows instructors to monitor student progress in real time and adjust instruction accordingly.

Administering Standards Approach 2 (Special class days). Another way to administer standards is to group them into 4-8 at a time and administer them to students every couple of weeks. This approach includes overlaps so that after the first set of standards, students will see a mix of old and new standards on each assessment. That is, the first assessment could cover standards 1-4, the second covers standards 1-8, the third covers 5-12, and continuing. Students only need to master each standard once, allowing them to "ignore" a standard if they had already mastered it. This rolling structure reinforces prior learning while gradually introducing new material, supporting both retention and mastery over time.

4.2 Grading Standards

In SBG, assessments are quickly noted as correct, incorrect, or something in between. The objective is two-fold: pass students only when they have demonstrated mastery, and minimize the time spent grading to provide feedback as quickly as possible.

Grading Approach 1 (Grading on a Binary Scale). In this approach the instructor grades an assessment as pass/fail. With each standard containing a few questions, students have to get all of them correct to receive a "pass" designation, which demonstrates complete mastery of the material. This method is particularly effective for reinforcing high expectations and clarity in what constitutes success.

Grading Approach 2 (Grading on a 3-Tiered Scale). In this approach the instructor grades an assessment as pass/half-pass/fail, corresponding to 1/0.5/0 points. This method provides a middle-ground to students who demonstrate mastery of most of the material but still have small mistakes present in their work. Assessments that receive partial credit (i.e., a half-pass) can be re-attempted to improve the score, though students also have the option to retain their original score if they prefer. These scores contribute to the standards portion of their final grade, allowing flexibility while still encouraging mastery through revision.

Grading Approach 3 (Grading on a 5-Point Scale). Another approach is to grade the students on a 5-point scale, with each level indicating more mastery of the material. For instance, students would receive: 4 points if completely correct, 3 points if minor errors are present but show thorough understanding, 2 points if they show basic understanding but are incorrect, 1 point if there are major misconceptions, and 0 points if very little attempt was made. This method allows for accommodating different degrees of understanding.

4.3 Incorporating Standards into Overall Course Grade

Typically, standards are one of the largest portions of the final grade, but other categories are given enough weight that students will be serious about practicing their skills in those assignments.

10:6 SBG in Technology Courses

■ **Table 1** Final Grade Approach 1.

Standards	Homework	Projects	Final Exam	Professional Development
55%	10%	10%	20%	5%

■ **Table 2** Final Grade Approach 2 (Before Final Exam).

	A	B	C	D	F
Participation (pre-class and in-class)	80%	70%	60%	50%	–
Homework (out of 20)	18	16	14	10	–
Projects (out of 12)	11	9	8	6	–
Standards (out of 20)	18	16	14	10	–

■ **Table 3** Final Grade Approach 2 (After Final Exam).

Grades before final:	A	B	C	D	F
Increase a letter grade	–	$S \geq 90$	$S \geq 85$	$S \geq 80$	$S \geq 75$
Maintain the letter grade	$S \geq 70$	$90 > S \geq 60$	$85 > S \geq 50$	$80 > S \geq 40$	$75 > S$
Decrease a letter grade	$70 > S$	$60 > S$	$50 > S$	$40 > S$	–

Final Grade Approach 1 (Percentage Determination). In this approach, standards form a significant portion of the student's final grade in the course, typically ranging from 35 to 60% of the final grade. For instance, students who pass 27 out of 30 standards (or earns 27 out of the 30 points when allowing for half-passes) receive a 90% for that portion of their grade. An example of how the breakdown might look can be seen in Table 1.

Final Grade Approach 2 (Category Minimums). In this approach, students must meet minimum performance thresholds in each grading category – such as participation, homework, projects, and standards – to determine their standing before the final exam. This method ensures that students demonstrate consistent effort and competency across all areas of the course, rather than relying on strong performance in just one category to compensate for weaker areas. The minimum requirements for each letter grade are shown in Table 2.

Once the pre-final grade is established, the final exam score (denoted as S) is used to determine whether the student's overall grade increases, remains the same, or decreases. This approach encourages students to continue engaging with the material through the end of the course, while also providing a safety net for those who have demonstrated consistent performance. The grade adjustment criteria based on the final exam score are outlined in Table 3.

5 Managing Assessments and Retakes

With each course having 20+ standards that students can re-attempt multiple times, it is important to have an organized management system to track student progress. Most of the instructors used Excel sheets to determine which students had passed (or attempted) each assessment and how many attempts each one took. This was beneficial as instructors could update the learning management system once per week instead of updating it every time a student attempted a standard.

5.1 Scheduling Retakes

A significant aspect of standards-based grading is the opportunity for students to re-attempt standards they have not yet passed. This helps promote a growth mindset by encouraging them to learn through revisions. To manage workload and to encourage students to reflect on their work and the material, the authors developed several methods for managing retakes and for creating new question sets for each retake. As with planning for the initial assessments and grading, readers are encouraged to find a combination of approaches that works best for their class and style.

Retakes Approach 1 (once per week). Students are limited to re-taking each standard once per week during office hours or during class. This approach helps control both the frequency of grading and question creation. It also encourages students to prepare more thoroughly before each attempt, knowing they have a limited opportunity.

Retakes Approach 2 (unlimited). Students are allowed as many retakes as they want, as often as they want. While this approach increases the instructor's workload in terms of creating and grading additional assessments, it empowers students to take full ownership of their learning by choosing when they are ready to demonstrate mastery. However, it is strongly recommended that students re-work their mistakes and reflect on their misunderstandings before attempting a retake. This encourages meaningful learning and prevents students from repeatedly retaking assessments in the hope of stumbling upon the correct answer without addressing the underlying concepts.

Retakes Approach 3 (certain days). Retakes are allowed only at specific times during the semester – such as designated “catch-up” days during class or open retake sessions near the end of the term. This model provides structured opportunities for reassessment while helping students manage their workload and avoid falling too far behind. By setting clear deadlines, this approach can help students prioritize and plan their study time more effectively.

Additional logistical considerations for managing retakes that include using office hours were the increased traffic for office hours and the need to proctor the re-attempts. Offering frequent retake opportunities often resulted in busier office hours for instructors, particularly later in the week. It was not uncommon to see a handful of students at a time working on re-attempts. The authors found it necessary to supervise the retakes to prevent academic dishonesty, sometimes by utilizing a nearby empty classroom.

5.2 Creating Retakes

To prevent students from memorizing the previous week's questions, new versions of each standard were created for every attempt. While this process can be time-consuming, especially during the initial transition to standards-based grading, it significantly reduces the workload in subsequent semesters, as a large bank of assessment versions is already available. The authors employ several methods to efficiently generate a wide variety of standards-aligned questions. For example, some faculty use Python scripts to automatically generate LaTeX code for customized questions, and R scripts to produce coordinated random data sets. This approach enables the creation of a nearly infinite number of question variations that still target the same learning objectives. The goal is to provide each student with a unique assessment opportunity for every retake, without altering the core structure or intent of the original standard.

Question Generation Approach 1 (homeworks). Some instructors build questions from homework assignments. This approach encourages the students to see the connection between practicing the application of course knowledge in the homework with the standards and their assessment. This also incentivizes students to not rely on AI when completing homework, as they need to actually understand the work they are doing for practice. Additionally, this method allows instructors to reinforce key concepts by reusing familiar problem structures, helping students recognize patterns and deepen their understanding. Homework-based questions can also be adapted to include slight variations, making them suitable for multiple assessment versions.

Question Generation Approach 2 (lecture notes). Some instructors build questions from lecture notes, using examples discussed in class as the basis for the assessment questions. This approach encourages students to be more present in class while also reinforcing the importance of active engagement during lectures, as students who take detailed notes and participate in discussions are better prepared for assessments.

Question Generation Approach 3 (previous exams). This approach maximizes past work by mining old tests for questions. Reusing and refining questions from previous exams not only saves time but also allows instructors to improve question clarity and alignment with current standards. This method supports the creation of a robust question bank, which can be especially useful for generating multiple versions of assessments for retakes.

Additional logistics considerations for generating assessments are the need to print the assessments and the amount of paper being used. Students request certain assessments (in class or in office hours) and the instructor pulls up their master document and copies the requested standards to print out. It is necessary to plan for the time to print as well as having the paper supply.

Overall, having an effective management system for standards-based assessments is important in documenting student progress and staying organized with all of the different versions being created. The effort invested in creating multiple versions of standards is crucial for ensuring that retakes accurately reflect student mastery of the material. The time spent grading does not noticeably increase though, as fewer exams are being administered. To streamline grading, especially given the binary nature of standards-based assessment, questions can often be designed for quick evaluation. When students retake assessments, some instructors grade them immediately after completion to provide instant feedback. However, in most cases, retake papers are returned the following week, as only one version is created per week, allowing the same version to be reused with other students. Occasionally, when instructors are able to grade student work as it is turned in, if a student is close to a correct answer, a small hint may be offered to guide them toward a successful revision.

6 Results

After two years of implementing standards-based grading in multiple sections, the authors have found several promising results as reflected in their retake statistics, student feedback, and classroom experiences.

One concern with allowing retakes is grade inflation. However, the authors did not observe this in their transition to SBG. The pass/fail nature of SBG also raises concerns of artificially lowering the grades. However, this was also not their experience. In fact, these two concerns

■ **Table 4** Historical Grades for Computer Security Course.

	18-19	19-20	20-21	21-22	22-23	23-24	24-25
median score	75.1	77.1	82.5	78.8	64.5	74.7	77.8
number of students	21	17	28	41	41	44	32
take-home?	no	no	yes	yes	no	no	no
standards?	×	×	×	×	×	✓	✓

tend to balance out. For example, in the Computer Security class, the same instructor has continuously taught 1-2 sections of the course for last seven years. The median final exam grade, shown in Table 4, has remained fairly consistent between the course prior to the implementation of standards and the course since standards have been added.

Looking at things more closely, pre-pandemic exam averages were 75 and 77. During the pandemic, the instructor switched to take-home, open-notes exams, with an attendant rise in scores. In 2022, exams were back in person, and there was a dramatic drop in scores. The instructor saw this drop across other courses as well. For the last two years, exams have been in-person and the course has used SBG. Scores recovered to their pre-pandemic levels. The instructor has noticed a drop in the preparedness of students pre-pandemic to post-pandemic. SBG may have offset this drop.

This analysis provides confidence that using standards is not hindering students, nor is it inflating grades. But the authors also see this as evidence that supports greater confidence that students are mastering the learning outcomes since a large portion of the final grade is based on passing standards.

One instructor had standards in all three of their courses (an introductory CS course, an upper-level CS course, and an upper-level math course) and noted that 36.2% of the first attempts to pass a standard were correct and an eventual pass rate of 88.7%. Students averaged 2.3 attempts before getting it right or giving up.

In Fall 2024, one instructor recorded 63 students making 4684 individual standard attempts, of which 1735 (37%) were correct attempts. There was a 34.5% initial pass rate with an eventual pass rate of 88.1%. Students averaged 2.2 attempts until correct or giving up. The maximum number of attempts for one standard by one student was 11.

End of course evaluations, administered by the university, include the option for students to provide written feedback on the course. There were many comments related to the use of standards. Many students appreciated the use of standards, with comments such as *“I liked how the standards allowed for an opportunity to better learn and grow”* or *“having these little moments to evaluate that add up are great”*. Some students did not like standards commented that they *“disliked the use of standards seeing it is difficult for me to code on paper”*. Seeing this feedback from past semesters, the authors have made a point in current courses that this is an opportunity to practice a skill they are not strong at, and that allowance is made for simple mistakes that a compiler would note and students would be able to quickly correct. Additional feedback from students can be found in the Appendix.

7 Lessons Learned

Overall, students appreciated the opportunity to retake standards as many times as they needed to show that they had mastered the material. However, some students disliked the amount of assessments that were given, their pass/fail nature, and the fact that the failed assessments piled up for them in multiple classes.

10:10 SBG in Technology Courses

Some instructors found that the pass/fail system was discouraging for beginning students who demonstrated that they knew most of the material but had not completely mastered it. Because of this, some instructors adopted a “half-pass” designation. Grading everything with this system allowed instructors to provide more nuanced feedback and encourage students who showed partial understanding to attempt retakes and improve their grades.

Some authors found that, while students don’t seem to study for the standards, especially in the entry-level courses, they do request to do retakes which eventually gets them to study. There is a high need to encourage students, sometimes in every class, to retake standards, announcing when those opportunities are to remind them to plan for the next chance and avoid procrastinating their retakes. There can be a negative impact on commuters, athletes, and other students whose schedules don’t align with the instructor’s availability. Consideration should be given to find a way to allow students the time to retake rather than being penalized for having a different schedule. One option to accommodate such schedules is to find class time where students can do retakes, such as if there is time after completing new standards or the odd day in the semester when it is not conducive (such as around breaks) to introduce or test on new material.

The authors acknowledge that standards-based grading may not be suitable for all courses, teaching styles, or learning environments. Alternative grading approaches can be equally effective depending on the course objectives and instructional context. For example, some of the authors have successfully implemented specifications grading in a general education math course, yielding positive outcomes. Another promising approach is ungrading or cooperative grading, which may be particularly well-suited for seminar-style or discussion-based courses. Exploring the effectiveness of these and other alternative grading models remains an important area for future research.

8 Discussion

Overall, the authors appreciate this grading approach because it reduces the pressure on students to master all the material by a specific deadline – such as a major midterm – instead allowing them to demonstrate understanding over time. It also encourages students to engage with instructor feedback and revisit material they initially struggled with in order to earn a passing grade on an assignment.

Generative AI is having a profound impact on the way students approach their coursework. This impact may be positive as informal tutors, or even as formal tutors as in the case of GPT tutors for calculus courses among others [8, 1]. Many teachers are, however, familiar with the negative impacts of AI disincentivizing study and mental work. When the learning objectives in a course can be expressed as a set of problem-solving skills, such as in many technical courses, the authors highly recommend the use of in-class assessments to combat this.

While a strict mastery or standards-based grading model typically excludes homework, the authors believe that eliminating homework entirely would not serve most of their students well. Despite the possibility of students finding ways to cheat, homework remains a valuable form of practice that provides opportunities for feedback. However, it must carry enough weight in the overall grade to incentivize completion; if the value is too low, many students simply won’t do it.

This grading model tends to benefit mid-range students the most. High-performing students generally succeed regardless of the grading system, while students who typically struggle to earn a C may now be able to achieve a B or even an A. Lower-achieving students

will still face challenges, though more of them may ultimately pass under this system. Additionally, it offers meaningful support for students who experience significant disruptions during the semester. For a student with a solid academic foundation, this approach allows them to recover and still perform well, rather than being penalized for circumstances beyond their control. It also encourages self-regulation, as students are responsible for monitoring their progress toward mastery and deciding when to reassess. The authors have found SBG instills a growth mindset, prevents cheating, rewards study, and, most importantly, has increased student learning as discussed in the results. Future work will focus on collecting more comprehensive data to evaluate the long-term impact of standards-based grading on student outcomes, as well as gathering student perceptions to better understand their experiences with this model.

References

- 1 Joel Alvarez. Evaluating the impact of ai-powered tutors mathgpt and flexi 2.0 in enhancing calculus learning. *Jurnal Ilmiah Ilmu Terapan Universitas Jambi*, 8(2), 2024. doi:10.22437/jiituj.v8i2.34809.
- 2 J B. Collins, Amanda Harsy, Jarod Hart, Katie Anne Haymaker, Alyssa Marie Hoofnagle, Mike Kuyper Janssen, Jessica Stewart Kelly, Austin Tyler Mohr, and Jessica OShaughnessy. Mastery-based testing in undergraduate mathematics courses. *Primus*, 29(5):441–460, 2019. doi:10.1080/10511970.2018.1488317.
- 3 Matt Bower, Jodie Torrington, Jennifer WM Lai, Peter Petocz, and Mark Alfano. How should we change teaching and assessment in response to increasingly powerful generative artificial intelligence? outcomes of the chatgpt teacher survey. *Education and Information Technologies*, 29(12):15403–15439, 2024. doi:10.1007/s10639-023-12405-0.
- 4 Melanie Butler. Alternative grading systems and student outcomes: a comparative analysis of motivation, enjoyment, engagement, stress, and perceptions of final grades. *Assessment & Evaluation in Higher Education*, pages 1–11, 2025. doi:10.1080/02602938.2025.2475068.
- 5 Robert Campbell, David Clark, and Jessica OShaughnessy. Introduction to the special issue on implementing mastery grading in the undergraduate mathematics classroom. *Primus*, 30(8-10):837–848, 2020. doi:10.1080/10511970.2020.1778824.
- 6 Debby RE Cotton, Peter A Cotton, and J Reuben Shipway. Chatting and cheating: Ensuring academic integrity in the era of chatgpt. *Innovations in education and teaching international*, 61(2):228–239, 2024. doi:10.1080/14703297.2023.2190148.
- 7 Stephen H Edwards, David L Largent, Ben Schafer, Debarati Basu, Amy Csizmar Dalal, Dan Garcia, Brian Harrington, Kevin Lin, Dan S Myers, Christian Roberson, et al. Developing a playbook of equitable grading practices. In *Proceedings of the 2024 on ACM Virtual Global Computing Education Conference V. 2*, pages 283–284, 2024. doi:10.1145/3649409.3691071.
- 8 Ibrahim AH El-Shara, Ahmad AS Tabieh, and Sahar YA Abu Helu. The effect of using matgpt on mathematical proficiency among undergraduate students. *International Journal of Information and Education Technology*, 15(4), 2025. doi:10.18178/ijiet.2025.15.4.2284.
- 9 Benjamin T Fine. Competency and equity driven grading system for computer science curriculum. In *Proceedings of the 2024 on Innovation and Technology in Computer Science Education V. 1*, pages 311–317, 2024. doi:10.1145/3649217.3653564.
- 10 Urfa Khairatun Hisan and Muhammad Miftahul Amri. Chatgpt and medical education: A double-edged sword. *Journal of Pedagogy and Education Science*, 2(01):71–89, 2023. doi:10.13140/RG.2.2.31280.23043/1.
- 11 Danielle L Iamarino. The benefits of standards-based grading: A critical evaluation of modern grading practices. *Current Issues in Education*, 17(2), 2014. URL: <https://cie.asu.edu/ojs/index.php/cieatasu/article/view/1234>.

- 12 Christian Jørgensen and Helen Bråten. Can ‘passed with distinction’ as a new grading scale favour the transition towards formative assessment. *Nordic Journal of STEM Education*, 3(1):7–11, 2019. URL: https://bio.uib.no/te/papers/Jorgensen_2019_Can_passed_with_distinction.pdf.
- 13 Carolina E Kuepper-Tetzel and Paul L Gardner. Effects of temporary mark withholding on academic performance. *Psychology Learning & Teaching*, 20(3):405–419, 2021. doi:10.1177/1475725721999958.
- 14 Drew Lewis. Impacts of standards-based grading on students’ mindset and test anxiety. *Journal of the Scholarship of Teaching and Learning*, 22(2):67–77, 2022. doi:10.14434/josotl.v22i2.31308.
- 15 Brendan McCane, Claudia Ott, Nick Meek, and Anthony Robins. Mastery learning in introductory programming. In *Proceedings of the nineteenth australasian computing education conference*, pages 1–10. ACM, 2017. doi:10.1145/3013499.3013501.
- 16 Claudia Ott, Brendan McCane, and Nick Meek. Mastery learning in cs1-an invitation to procrastinate?: Reflecting on six years of mastery learning. In *Proceedings of the 26th ACM Conference on Innovation and Technology in Computer Science Education V. 1*, pages 18–24. ACM, 2021. doi:10.1145/3430665.3456321.
- 17 Daniel E Rohe, Patricia A Barrier, Matthew M Clark, David A Cook, Kristin S Vickers, and Paul A Decker. The benefits of pass-fail grading on stress, mood, and group cohesion in medical students. In *Mayo Clinic Proceedings*, volume 81, pages 1443–1448. Elsevier, 2006. URL: [https://www.mayoclinicproceedings.org/article/S0025-6196\(11\)61250-0/abstract](https://www.mayoclinicproceedings.org/article/S0025-6196(11)61250-0/abstract).
- 18 Kevin R Sanft, Brian Drawert, and Adam Whitley. Modified specifications grading in computer science: preliminary assessment and experience across five undergraduate courses. *Journal of Computing Sciences in Colleges*, 36(5):34–46, 2021. URL: <https://dl.acm.org/doi/abs/10.5555/3447307.3447310>.
- 19 Carol A Tomlinson and Jay McTighe. *Integrating differentiated instruction & understanding by design: Connecting content and kids*. Ascd, 2006. URL: <https://www.ascd.org/books/integrating-differentiated-instruction-and-understanding-by-design>.

A Sample Standards

A.1 For a Java Course

1. Variables	9. Variable Scope	17. Java Swing
2. Expressions	10. Control Flow	18. String Manipulation
3. I/O	11. Conditionals	19. Regular Expressions
4. Typecasting	12. More Conditionals	20. ArrayList
5. Wrapper Classes	13. Indefinite Iterations	21. Object Oriented Thinking
6. Random Class	14. Definite Loops	22. Inheritance
7. Math Class	15. Arrays	23. Composition
8. Simple Arrays	16. File I/O	

A.2 For an Introductory Data Science Course

1: Identifying Type of Data	11: Discrete Distributions
2: Binary, Hexadecimal & Other Bases	12: Continuous Distributions
3: Calculating Mean/Median/Mode	13: Central Tendency
4: R Basics	14: Describing Spread
5: Vectors in R	15: Describing Skewness
6: Assessing Vector Elements & Logical Operators	16: Quantitative Visualizations
7: Factors in R	17: Qualitative Visualizations
8: Dataframes in R	18: Central Limit Theorem
9: Exploratory Data Analysis	19: Confidence Intervals
10: Tables and Aggregate Function	20: Hypothesis Testing

B Sample Questions

B.1 For a Java Course

A selection of assessment questions used with the standards in Appendix A.1.

Standard 1. Add code below that creates a decimal number variable `x` equal to 8.5 and a whole number variable `y` equal to 7. Print both values in a user-friendly manner.

Standard 2. Given the following variable declarations, add code below that stores the sum of these numbers in variable `s` and the product in variable `p`. Print all four values in one user-friendly statement.

Standard 10. Draw a control flow diagram for the problem of determining if an integer is even or odd.

Standard 15. For a two-dimensional array `a` of integers that is 4 rows and 8 columns long, using loops, print out every item except for the last two columns.

B.2 For an Introductory Data Science Course

A selection of assessment questions used with the standards in Appendix A.2.

10:14 SBG in Technology Courses

Standard 5. Create a vector “x” containing the values 8, 6, 7, 5, 3, 0, 9 and determine what the output would be for $x[x < 6]$ and $x < 6$.

Standard 6. Given a vector “x” containing the values 8, 6, 7, 5, 3, 0, 9 write code using logical operators to determine which values are either equal to 6 or less than 3.

Standard 13. Describe how the central tendency statistics will look when dealing with a left-skewed distribution. Visualize it.

Standard 19. You decide that you want to create a 95% Confidence Interval for the mean scores on a Calculus II Exam. You sample 16 people and determine the average score was 72. It was previously determined the population standard deviation is 12. What is the 95% CI (assume $z=1.96$) and what does it mean in “real-world” terms?

C Student Feedback

A selection of quotes students gave the instructors in their course feedback:

- “I really like that the class assessments are able to be done as many times as needed. Going back in really helped me learn the subject and retain the information better.”
- “As a student who sometimes does not perform well on traditional exams due to anxiety, having these little moments to evaluate that add up are great.”
- “It gave me the opportunity to understand the material and hold myself accountable for learning, rather than have the pressure of a traditional grading system.”
- “I disliked the use of standards seeing it is difficult for me to code on paper.”
- “I think it is so beneficial to be able to learn more about work and do it until you get it right. It drills it in your brain.”
- “...if you didn’t pass you would have to retake which made it hard to keep up with not only this class but others.”