Early Experiences with Specifications Grading in Introductory CS Courses

Stephen H. Edwards
Computer Science
Virginia Tech
Blacksburg, VA, USA
edwards@cs.vt.edu

Manuel A. Pérez-Quiñones
Software and Information Systems
UNC Charlotte
Charlotte, NC, USA
perez.quinones@charlotte.edu

Adrienne Decker
Engineering Education
University at Buffalo
Buffalo, NY, USA
adrienne@buffalo.edu

Bob Edmison
Computer Science
Virginia Tech
Blacksburg, VA, USA
bedmison@vt.edu

Audrey Rorrer
Computer Science
UNC Charlotte
Charlotte, NC, USA
audrey.rorrer@charlotte.edu

Anmol Shukla
Computer Science
Virginia Tech
Blacksburg, VA, USA
ano22@vt.edu

***Abstract*—This innovative practice paper describes our experiences with alternative grading practices in introductory computing courses and two large public universities in the United States. Computing classrooms often use traditional grading practices involving allocating points to assignments, deducting points for mistakes and tardiness, and combining assignment scores using a weighted average to determine grades. Recent research suggests that these practices may diminish achievement, discourage students, and suppress effort to such an extent that they are considered by some as detrimental.**

We approach our work as an exploratory case study, without predefined research questions or hypotheses. Our experiences began with the adoption of specifications grading. We outline the grading scheme applied to traditional programming assignments and exams/quizzes, and discuss the initial integration of these schemes with conventional auto-grading tools. We delve into student perceptions of alternative grading, their utilization of flexible deadlines, and resubmission opportunities. We conclude with a discussion of two challenges encountered during our exploration: student acceptance of a novel grading form, and the adaptation of tools designed for traditional grading to support alternative grading mechanisms.

Our early exploration aims to inspire further research on the use of alternative grading in computing. It is clear from our observations that simply implementing the practices does not ensure the equitable and inclusive outcomes that can be achieved with these practices. If students are not prepared to use these practices, they find them difficult to understand and can feel that they are not being treated fairly. Additionally, we wish to foster a community of practice to assist faculty members exploring these changes, with the goal of creating more equitable and inclusive classrooms.

***Index Terms*—Computer science education, Student assessment, Equitable grading, Specifications grading, Grading practices**

I. Introduction

As educators, we often state that we believe that the most important outcome of our courses is student learning. However, in the current structure of most higher education classrooms, the final course grade, typically determined by a point system, is what students ultimately leave with. Students often find themselves racing to accumulate enough points to achieve their desired grade in a course. These points are usually averaged scores from various class components such as assignments, quizzes, laboratory work, and exams. By the time students arrive at their university, they have developed strategies to maximize points by understanding the rules of a course, its grading structure, and balancing these against other competing demands in their lives.

To shift students' focus away from the pursuit of points and towards learning objectives, we need to radically change our grading practices. This shift towards alternative grading practices comes with costs for both instructors and students. These costs include the time required for instructors to transition and educate students about new processes and procedures, and a mental shift for students regarding their goals and behaviors. In addition to time, instructors must navigate the now ubiquitous infrastructure in higher education, such as learning management systems, auto-graders, and e-books.

This paper explores the challenges of transitioning to alternative grading practices from the perspectives of two different universities. While the changes made to the courses to adopt alternative grading strategies were very similar, the experiences at the two universities and reactions from students varied significantly. Both institutions transitioned from points-based grading to specifications grading, eliminated deadlines, allowed resubmissions of work to earn higher scores, and lowered the stakes on assignments. However, these changes resulted in slightly different student behaviors and reactions at the two institutions.

II. Background

Often referred to as alternative grading practices, many of the practices we will discuss in this paper help level the

This material is based upon work supported by the U.S. National Science Foundation under Grant Nos. 2235337, 2235643, and 2235644.

playing field in courses for students of varying backgrounds and are often also referred to as equitable grading practices.

A. Equitable Grading

Most Computer Science (CS) classrooms use traditional grading practices where points are allocated to assignments, mistakes result in point deductions, and assignment scores are combined using some form of weighted averaging to determine final grades. While these practices have served most instructors well, these practices can *reduce achievement, discourage students, and suppress effort* [1], [2]. Some even go as far as to say these practices could be *toxic* [3]. The main problems center around: using zeros for missing work; averaging scores throughout the academic term; and using heavily weighted high-stakes tests or project assignments where one assignment's score can make the difference between achieving a passing grade or needing to repeat a course. Using grades punitively is an ineffective teaching strategy because it increases anxiety and stress, yet these toxic grading practices are virtually ubiquitous throughout higher education.

According to Feldman [4], traditional grading includes a component that evaluates student's behaviors, often including timeliness, effort, and other behavioral measures. These metrics, which are not based on content knowledge or learning outcomes, are often influenced by implicit biases, personal factors outside of the classroom (e.g., part-time employment), and by family life situations (e.g., students who are caregivers). Students in these situations are disproportionately students from marginalized communities (e.g., low-socioeconomic background, first generation, and by proxy, students of color).

B. Specifications Grading

Feldman [4], Rapaport [5], and Nilson [6] propose alternative grading practices that advocate for a reduced grading scale. Feldman [4] suggests that a 0-100 scale, with 0-59 being failure, tilts the scale towards failure. Feldman advocates we use "minimum grading", meaning that the minimum score that a student could obtain for a no submission is a 50 instead of a 0. Nilson [6] suggests a pass/fail (satisfactory/unsatisfactory) grading scale with opportunities for students receiving a fail grade to resubmit work, thereby providing additional opportunities for practice. Rapaport's triage grading [5] provides a third option, such that any item to be graded gets one of three grades: full credit if it is clearly/substantially correct (i.e., 3 points); minimal credit if it is clearly/substantially incorrect (i.e., 1 point); emerging credit if the item is neither of the above (i.e., 2 points); zero credit if the item does not exist (i.e., 0 points).

Partial credit grading within a 0-100 scale has several disadvantages as recognized by Feldman, Nilson, Rapaport and others [3]–[7]. Students feel they can argue for a few more points after the fact without doing additional work, or worse, view receiving points as a game that needs to be won as opposed to it being a reflection on their learning [5], [6].

C. Aggregating Points

Averaging scores across assignments using arbitrary weighting schemes for different types of assignments can also be problematic [6]. Often, courses assign a certain weight to in-class assessments and a lower weight to out-of-class assessments. This places a burden on students to perform in high-stakes, timed situations to demonstrate their learning. In addition, the aggregation of points to create an overall grades loses valuable information about what knowledge the student does or does not have.

D. Resubmissions and Regrading

Allowing resubmission of an assignment lowers the stakes of the assignment, provides students with additional opportunities to practice and complete the assignment, and more closely matches the learning outcomes of the assignment or course [6]. This resubmission and regrading is likely most effective when paired with quality feedback about the original score. With scoring like triage grading, a student can see what pieces of an assignment they were able to complete successfully and which they were not. As opposed to gaming the system for points, more credit is earned when the student successfully completes the sections they did not earn full credit on. The goal is to support more learning in the process.

III. UNC Charlotte Study

A. Context

UNC Charlotte, a state university classified by the Carnegie Classification[1] as "Doctoral Universities: High Research Activity," has a student body of approximately 30,000 students. The urban campus is located in a large city, and the student body is primarily residential. The undergraduate profile is characterized as "Selective, Higher Transfer-In." The CS major is housed in a separate College of Computing.

At UNC Charlotte, we adopted specifications grading to make education more equitable and to minimize the impact of external factors in the classroom (e.g., COVID-19 and social inequities). We explored the use of specifications grading in a senior technical elective for two semesters and then adopted it for a section of our Data Structures and Algorithms course. This report covers two offerings of this course in Fall 2021 and 2022.

The Data Structures and Algorithms course, the third course in the introductory sequence for CS Majors, begins with a review of arrays and quickly moves to linear data structures, algorithm analysis, trees, and concludes with a presentation (but not much practice) of graphs and graph algorithms. The course utilized OpenDSA [8] as its online textbook, CodeWorkout [9] for drill-and-practice exercises, and Web-CAT [10] for grading larger projects. All three systems support auto-grading of assignments. We also had three exams that were graded using a more traditional points-base approach to give a percentage score.

[1] https://carnegieclassifications.acenet.edu

B. Grading

At UNC Charlotte, we adopted the Satisfactory/Unsatisfactory (S/U) grading scale and added one more category, *Not Submitted*, to distinguish between students who were not participating and those who were attempting but struggling. Thus, students would receive a *Not Submitted* for no attempts, which was different from students who attempted an assignment but did not perform well enough to reach the Satisfactory level.

It is worth noting that the automated graders used did not support this type of triage grading. The result of evaluation is typically a percentage-based score. We had to map such a percentage in an assignment to an S/U grade. To accomplish this, we established percentage equivalencies. For CodeWorkout, for example, we used 80% as the cutoff for Satisfactory. In the LMS (we use Canvas from Instructure), we defined a grading scale that essentially had Satisfactory for scores $\geq$ 80%, Unsatisfactory for scores below 80%. The Canvas gradebook showed grades using Satisfactory/Unsatisfactory for each submission, but it did not provide a reliable way to compute and display the final grade.

Based on the end-of-term student evaluations, students appreciated the flexibility that this grading approach afforded them. Several students commented positively about this new grading approach. One stated, "I like the grading policy" (Fall 2021). Another student called the grading scheme "unconventional" but considered it to be "pretty effective in getting the grade that you deserve [...]" (Fall 2022). Another student called it "nice compared to an average class. Easy to get a better grade for the most part and the work is not above and beyond" (Fall 2022).

Several students explicitly mentioned how this approach of grading reduced stress in the class. One stated that the approach was "Incredibly lenient and attempts to make the class as stress-free as possible" (Fall 2021). Another student said that the grading system was "fantastic" and "Wasn't stressful" (Fall 2021). A third student called it "excellent" and stated that it made them "less stressed about the class and more eager to learn about the topic" (Fall 2022).

A few students explicitly stated that the approach encouraged more learning. One highlighted the "Unique grading system that encourages and motivates" (Fall 2021). A second student "loved the grading style" because "it promoted learning at your own pace" (Fall 2022). Finally a third student pointed out the trade-off between learning and getting an A in the class by stating: "I like method of grading work because it promotes actually learning the material instead of just trying to get an A in the class" (Fall 2022).

The literature on alternative grading discusses the elimination of conflict between the professor and the student regarding grading [6]. One student noticed this and commented: "Very understanding and gives students many opportunities to succeed. The PROFESSOR really wanted us to learn the material rather than just pass tests" (Fall 2021). This professor has always prioritized student learning and did not alter their teaching style from their previous 20+ years of experience. However, the perception of concern and care conveyed through this grading mechanism was clearly recognized by the students.

In future semesters, UNC Charlotte plans to adopt a grading scale similar to the one used by Virginia Tech (see next section) where there is a finer separation between exceeding and meeting expectations. This should allow students to make decisions regarding how much extra work to put in each assignment.

C. Deadlines

At UNC Charlotte, we relied entirely on automated grading. In fact, we had minimal support from teaching assistants—we had none in the first semester, and only partial support in the second semester. Auto-grading did alleviate our burden in terms of grading resubmissions, which was an important concern since students could resubmit unsatisfactory work at any time. Auto-grading supports this without requiring additional staff resources and grades all attempts equally, regardless of when they were submitted. But auto-grading is not always a possibility, thus considering how to handle the manual grading workload associated with resubmission attempts requires serious consideration before adopting some of these practices.

In Fall 2021 at UNC Charlotte, we allowed students to submit smaller assignments (OpenDSA and CodeWorkout) until the last week of the semester. This became problematic because students' missed work accumulated towards the end, and then they attempted to complete an overwhelming amount of work the week before finals.

Supporting resubmissions encourages continuous growth and learning. However, allowing what amounted to a free-for-all with late submissions clearly did not work in this particular situation. First, students still had to complete formative assessments in the form of exams at various points in the semester. These assessment instruments were not *moveable*, and thus students would approach these assessments without the necessary level of practice or exposure required for them to demonstrate mastery of the content.

Second, there is always an amount of incremental knowledge acquisition that builds up over the semester. Attempting to do all the work at the end does not allow for that to happen. Third, it was much more challenging to answer questions in the last two weeks of the semester because students were asking questions about topics covered weeks earlier. It also made it pointless to explain common misunderstandings in, for example, week 3, and then have a student come at the end trying to clarify the misunderstanding that now became relevant because they were only then attempting a particular assignment.

For the Fall 2022 semester, the late deadline was set to coincide with the exam where the material was covered. So, OpenDSA and CodeWorkout assignments were due throughout the term to coincide with material covered in class, but the late submissions were accepted (without penalty) until

the night after the exam that covered that material. This strategy worked well and encouraged students to make steady progress. Students showed some confusion about having a "late" submission without a corresponding penalty. They had a hard time accepting the idea of submitting work late without being penalized. This observation is a reflection of how our educational system and the mindset of our students consider late work punitively.

Overall, the flexible deadlines were well received. "The expected due date vs actual due date worked well for me" (Fall 2021), commented one student. This approach encouraged students to continue working on the assignment: "The idea of having a due date but an extension to the next week was much more motivating than having a hard set deadline" (Fall 2021). Eliminating the late penalties was well received: "I liked the fact there were extended due dates with no late penalties" (Fall 2022). Another student described our strategy as an "excellent strategy" because it allowed the student to "complete work when [they] had time," adding that they "didn't have to rush to finish assignments and actually try and learn the material instead of just trying to get a grade" (Fall 2021).

D. Final Grades

To compute final grades at UNC Charlotte, we used the bundle idea as described by Nilson [6]. To get an A, for example, students needed to complete a minimum number of assignments in each category (OpenDSA, CodeWorkout, and Web-CAT) at the Satisfactory level. For a B, the number of Satisfactory assignments was slightly lower for each category, and so forth. This was organized in a table in the syllabus and explained in class on the first day. The explanation even included some literature background justifying to students why this approach was used and trying to reassure them that this was not intended to be a punitive approach to grading.

Due to the lack of support in the LMS for computing the final grade in the course using this alternative approach, students had to count the number of Satisfactory grades for each assignment type to figure out what grade they currently had. Because of the number of students expressing concern about not knowing their current grade, the professor sent email updates to each student with a calculation of their current grade throughout the term.

Some students did not like this new grading approach because it was not what they expected. One student commented: "Didn't really like the grading system of his class as it was reliant on how many assignments you got done instead of a traditional grading system" (Fall 2022).

1) Flexible Grading Not a Motivating Factor: At least one student saw the flexibility and encouragement for learning as counterproductive and possibly even encouraging procrastination, or at least not penalizing it. This student commented on how the professor needed to be "a little harsher." The student admitted "I did procrastinate so [the professor] should make some assignments open not as long to make sure people really do them" (Fall 2021).

Grade	Specifications	Traditional
A	32.8%	22.0%
B	26.9%	32.2%
C	23.9%	23.7%
D	7.5%	7.6%
F	9.0%	14.4%

TABLE I

GRADE DISTRIBUTION COMPARING THE SPECIFICATIONS GRADING SECTION OF DATA STRUCTURES AT UNC CHARLOTTE IN FALL 2022 WITH TRADITIONALLY GRADED SECTION OF THE SAME COURSE TAUGHT BY A DIFFERENT INSTRUCTOR.

Grade	Specifications	Weighted
A	32.8%	8.8%
B	26.9%	35.3%
C	23.9%	27.9%
D	7.5%	13.2%
F	9.0%	14.7%

TABLE II

GRADE DISTRIBUTION COMPARING SPECIFICATIONS GRADING AND TRADITIONAL GRADINGOF DATA STRUCTURES AT UNC CHARLOTTE IN FALL 2022.

2) Final Grade Based on Lowest Category (Bundles): More than one student did not like that their overall grade was determined by their "weakest link" (see Section IV-D1). One expressed: "Maybe find a way to grade differently so one category outside a bracket [bundle] doesn't resolve the entire grade" (Fall 2022).

Another student provided an insightful analysis of this approach while suggesting an alternative: "I think that the idea that your overall grade for the class is the lowest of the categories needs to change. For some individuals in the class, they are able to work hard and do well on assignments, projects, and textbook work, but if they aren't a good test taker, none of the rest of the work matters, and they are gonna get a bad grade. Maybe some sort of average for the 4 categories would be better, with certain grades for each category give point values, and if your point total decides your final grade" (Fall 2021). Averaging the four categories is also discussed in Section IV-D2. In future semesters we plan to explore an approach similar to the one discussed in Section IV-D2.

We compared the distribution of grades obtained in the Fall 2022 section using specifications grading to another section of the same course offered by a different instructor that employed a traditional grading scheme. Table I show the distribution of grades in both sections. Using the Wilcoxon Rank Sum test to compare these distributions, we determined there was no statistical difference between them (z=-1.52121746114833, p=0.12).

To ensure that the specifications grading approach did not penalize students, we computed final grades after the fact using the same weighted averaging grading scheme that was used in previous semesters. We then compared the distribution of grades obtained using these two approaches, Table II show the distribution of grades using each grading strategy for the same group of Fall 2022 students. Using the Wilcoxon Rank Sum test to compare these distributions, we determined there was no statistical difference between them (z = -0.313339780720256,

p = 0.75).

E. Challenges with Tools

1) Late Submissions in Tools: We faced a number of challenges trying to support our grading approach with existing tools that were not designed for these purposes. First, not all tools have the same support for late submissions. Some allow a due date and a separate "open until" date (i.e., late period) with optional penalties. This was implemented differently in the various tools we used, making it a challenge to keep dates accurate. Some tools simply blocked access to exercises after the last submission date. In the spirit of allowing students to continue practicing, tools should separate dates when they stop receiving submissions from dates used to calculate grades. Students should be able to practice even after the deadline when the grade is recorded.

Some of these challenges with tools were noticed by students. Because students had to count Satisfactory grades to figure out their current course grade, they sometimes misunderstood or mistrusted the grading system. One student commented simply: "Allow us to see our grades" (Fall 2022).

2) Challenges Mixing Specifications Grading with Percentage-based Grading: We encountered challenges with tool support due to their over-reliance on percentages as a grading measure. Tools that grade using a monolithic percentage score are more difficult to configure for specifications grading. This issue is similar to the problem of averaging scores mentioned in Section II, but it is now embedded internally in each assignment. A percentage grade does not differentiate between what a student gets right and what they get wrong. A specification grade should indicate individually which of the assignment's specifications have been met. Thus, percentage scores can obscure the fact that some parts of the assignment might be incomplete.

Ideally, we would like grading support systems to recognize multiple learning objectives per assignment. For example, an assignment in a Data Structures course might be associated with the use of object-oriented design, algorithm efficiency, and code documentation. We should be able to grade each learning objective separately in each assignment. This would allow us to assess each course learning objective at the end of the term. When tools reduce grades down to a single percentage, we lose the ability to tease apart the student's achievement with respect to the different learning objectives for the assignment and the course.

IV. Virginia Tech Case Study

A. Context

Virginia Tech, a state university classified by the Carnegie Classification as "Doctoral Universities: Very High Research Activity," has a primarily residential student body of nearly 40,000. Located in a rural setting, the university is characterized as "More Selective, Lower Transfer-In." The CS major is housed within the College of Engineering.

At Virginia Tech, we adopted specifications grading for reasons similar to those at UNC Charlotte. This approach not only improves equity and helps students minimize the impacts of external factors on their performance, but it also empowers students by shifting the emphasis from points to demonstrating mastery of learning objectives. We implemented this approach in our CS1 course, an introduction to Java programming. This course, which covers standard CS1 topics and aligns with the College Board's AP Computer Science A curriculum, is the first course taken by computer science majors and minors.

Like UNC Charlotte, this course at Virginia Tech uses OpenDSA to provide an electronic textbook. Each reading assignment from the e-textbook is tightly integrated with embedded practice exercises using CodeWorkout, which includes both syntax practice exercises and problem-solving activities. Students also have weekly lab assignments and out-of-class programming assignments approximately every two weeks, all of which are graded using Web-CAT.

In terms of summative assessments, the course previously included two midterm examinations and a final exam. To move away from high-stakes exams, we modified the course to use eight in-class quizzes, each lasting about 20 minutes. These quizzes were administered every two weeks. The goal was to reduce test anxiety and mitigate the risks associated with two large exams that are individually worth a large portion of the course grade. This change improved the effectiveness of summative evaluation, as each quiz focused on the learning objectives for a smaller two-week period, while allowing 60% more question time (eight 20-minute quizzes vs. two 50-minute midterms). The course retained a single high-stakes final exam.

B. Grading

We used specifications grading for all lab assignments and programming assignments. Instead of a two-valued scale, we opted for a four-valued scale similar to Rapaport's [5]. We used the EMRN scale [11], which we defined for students as follows:

- **(E) Excellent**: Exceeds expectations with no nontrivial errors.
- **(M) Meets Expectations**: Meets expectations by clearly demonstrating intended skills, with no significant gaps or errors, although some revision is needed.
- **(R) Revision Needed**: Partial understanding/skill is shown, but significant gaps or errors are present requiring revision.
- **(N) Not Assessable**: The work is too fragmentary or has too many gaps to assess whether there is understanding of the required concepts.

Each assignment had specific objectives stated in the assignment instructions, which were also echoed in feedback. We allowed students to resubmit or retry their lab work and programming assignments an unlimited number of times to demonstrate mastery of the required objectives. At the same time, meeting expectations required students to self-test their own solutions by writing their own software tests. Students who did not thoroughly test their work were expected to address this deficiency before receiving more feedback on

remaining objectives. The Canvas gradebook for the course displayed assignment grades using the EMRN category names.

While we used specifications grading for all lab and programming assignments, we combined it with more traditional points-based grading for weekly reading assignments, quizzes, and the final exam. Such a hybrid approach has significant limitations. However, it was necessary as a compromise in order to deploy specifications grading as we move towards a more integrated grading scheme.

1) EMRN is Preferable to Pass/Fail Grading: One lesson learned from this experience is that at Viriginia Tech, a four-valued grading scale was useful. Even when not meeting expectations, there is value in distinguishing between work where the student is making an effort and progress, versus missing work. At the same time, even when work is satisfactory, being able to distinguish between work that meets the bare minimum expectations and work that goes above and beyond is helpful. This allows the grading scheme to encourage students to achieve more and also to recognize when they do. The EMRN scale provides room for making these distinctions while still retaining the basic notion of meeting specifications.

2) Specifications Grading of Course Work Eliminates Point Seeking: We learned that one of the often-cited benefits of specifications grading—eliminating the common experience of students negotiating or arguing over specific point deductions—is readily observed. Converting to EMRN grading on lab and programming assignments virtually eliminated student questions or interactions about individual point deductions. The use of clear objectives in the assignments and referring to those objectives in automated feedback made shortcomings clear and easy to explain. Even when there were students who thought they were "close enough," they always had the option to revise their work and resubmit.

C. Deadlines

As advocated by Nilson [6], we embraced the idea that demonstration of learning objectives is more important than timing. Allowing students to revise and resubmit work that does not meet expectations follows from this idea, and while deadlines are important, they can conflict with the goal of allowing students to learn after failing to meet expectations. At the same time, deadlines also provide important benefits. Deadlines are important to set expectations for students and keep them on track. Especially for first-year university students who may not yet have developed the time management skills needed to succeed in the remainder of their college careers, setting progress milestones to combat procrastination can be important scaffolding. For large enrollment courses, deadlines are important tools for managing the workload on course staff in terms of providing timely feedback. Also, in terms of periodic assessment activities such as exams, deadlines on other coursework help ensure that students are appropriately prepared before the examination.

In trying to balance these concerns, we set assignment deadlines, but employed a "generous" late policy. Except for quizzes, students were given up to one week past the posted deadline to turn in late work, with the qualification that earning an (E) Excellent grade required turning the work in on time. However, students were encouraged to turn in late work within the one-week period in order to achieve a (M) Meets Expectations grade. We arrived at this approach after allowing students to resubmit work up until near the end of the course in a previous semester—an approach that resulted in students ignoring deadlines almost entirely. For quizzes, although there were no "late" attempts provided, the course grading policy dropped the lowest quiz grade in order to account for situations where students missed a quiz for any reason, or performed worse than expected.

1) Unintentionally Undermining Deadlines and Increasing Procrastination: Many students saw useful benefits to a generous late policy with minimal penalties, since it allowed them to accommodate various scheduling events or life challenges without the stress of worrying about the impact on their grade if they needed an extra day or two to complete an assignment. However, for a significant number of students, a generous late policy completely undercut the use of traditional deadlines: students saw the posted deadline as meaningless, since it carried minimal consequences if missed, and instead saw the late deadline as the "true" deadline.

In some cases, there were students who habitually turned in everything late. Allowing only a one-week late submission window prevented students who were habitually late from falling too far behind, but they were still a full week behind the rest of the class. With these students, and with others, students would simply wait until near the end of the late period to even start their assignment, and then when they found themselves struggling, would ask for an "extension," even though they had automatically received a one-week extension from the posted deadline already. This forced students with poor time management skills into a situation where they had no option and were simply unable to turn in work at all because time ran out during the one-week late period.

In the future, more work is needed on balancing these competing forces between using deadlines to help keep students from falling too far behind and to provide structure to help them combat their own time management limitations, while also providing meaningful ways to help them recover from failure or to rework and resubmit unsuccessful solutions once they have mastered the corresponding learning objectives. Other educators have described token-based systems or time banks to impose a "restricted resource" mentality on deadline extensions while still allowing flexible resubmission that are worth exploring.

2) Some Students Recognize the Opportunity Cost Associated with the Late Policy: Simultaneously, students who are more adept at time management appreciate the flexibility. In informal interviews conducted at the end of the semester, high-achieving students acknowledged that while the late policy can assist with time management or enhance the quality of their work, late submissions impose an opportunity cost. Postponing work beyond the deadline will encroach on their schedule

during the period when they should be working on the next assignment. To maintain pace, they must put in additional work after the deadline, not less. In other words, even though there is no point penalty associated with late submissions under our flexible policy, late work is not *free*. However, it requires a higher level of self-reflection and awareness for students to recognize this cost and incorporate it into their own planning.

D. Final Grades

When assigning course grades at Virginia Tech, we employed a hybrid strategy that combined EMRN grading for lab and programming assignments with traditional points for quizzes, the final exam, and reading assignments. Hybrid schemes pose definite challenges but were chosen in this case due to negative reactions from students to a version of Nilson's bundle approach [6] (as employed by UNC Charlotte) in a previous semester. In that prior semester, we employed EMRN grading on all reading assignments, quizzes, and the final, and then specified a minimum number of (M) Meets Expectations grades in each category of assignments for each grade level, with grades at the A or B level also requiring some level of (E) Excellent performances beyond (M) Meets Expectations.

1) Students Strongly Object to Grading on the "Weakest Link": When using Nilson-style bundling, students needed a minimum number of M or E grades in each assignment category to earn a specific final letter grade. However, this meant that if students met the required minimum for the number of quizzes meeting expectations, but fell short in lab work, or met the required minimum in labs but fell short in reading assignments, etc., their final letter grade would be determined by the type of work showing the lowest level of mastery.

From the teacher's perspective, this approach works effectively because students can rework and resubmit most assignments until they meet expectations. Summative assessments like quizzes or the final exam still play a significant role in determining the student's grade, but with effort, students can raise the level of performance on any other work by resubmission, within the constraints of the course calendar. However, to students, the perception was very different. Particularly among high-performing students, the idea that students must perform at a specified level *across the board* on all assignment types in order to earn a specific course grade was seen negatively. Students directly concluded that whatever area of the course they were performing "worst" in would determine their final grade, regardless of how well they did in other areas, without cognitively connecting this to the idea that raising their performance ratings in coursework was under their control through effort and resubmission.

2) Students View Score Averaging as a Strength, Not Weakness: In this vein, students also clearly saw the idea of numeric averaging of scores as a "strength" in traditional points-based grading, and that specifications grading was less desirable because it removed this aspect. Students expressed the idea that they rely on higher numeric scores being able to "cover" or pull up lower scores through averaging. Students habituated to traditional points-based grading *expect* averaging and view it as a tool: they view the ability to have stronger performance in one area "average out" weak performance in some other area as a crucial element in being graded "fairly.''

3) Hybrid Grading Systems are a Challenge: The reactions from students to bundled specifications grading were one of the primary reasons we retreated to a hybrid scheme during the most recent semester at Viriginia Tech. At the same time, hybrid approaches reintroduce many of the problems or limitations associated with traditional points-based grading. Hybrid approaches are a challenge because they may produce a grading system that exhibits the worst aspects of both styles.

In this semester, our hybrid approach included traditional points-based and EMRN assignments. To do this, EMRN-graded assignments were mapped to numeric scores, with E grades earning 100% credit, M grades earning 80% credit, R grades earning 50% credit, and N grades earning 0%.

Two students could put in noticeably unequal efforts and they could be rewarded the same grade, especially around the (M) Meets Expectations grade. Some students attribute this to the broad nature of EMRN grading and the relative difficulty of achieving the E vs. M grades.

4) Students Experienced Anxiety and Confusion when Specification Grading was Introduced in the Course: During our initial attempts at employing specifications grading, it became clear that merely providing a direct explanation of the grading approach and what it means is insufficient. While no one expects students to be familiar with a new grading strategy, this approach is so different from the traditional points-based approaches that students have lived with for their entire academic careers that more work is necessary to successfully communicate the ideas to students and get them to buy into how the system works. Beyond setting expectations, faculty members adopting new grading approaches need to embrace the idea that they are embarking on a significant cultural change in the classroom. Students may need significant and repeated discussion of the reasons for the change, the goals you are trying to achieve, and how they should think about assessment overall.

One of the biggest lessons learned is that failing to recognize these communication needs can lead to significant complications and negative reactions. Adopting radically different grading policies and practices can produce "culture shock," and active steps should be taken to address this. While retreating to a hybrid approach was partially in response to the significant culture shock we experienced on our first outing, it also helped provide clearer pathways towards deeper adoption of specifications grading by helping to identify the issues we need to address.

E. Challenges with Tools

At Virginia Tech, our challenges with educational tools were exactly as described in Section III-E. This is not surprising, since the same learning management system and the same external learning tools were used. However, it is clear that existing learning management systems and educational tools

heavily rely on the assumption that grades are communicated and stored as numbers. Existing learning tools do not provide effective ways of grading using non-points-based schemes, or structuring feedback in a way that is not driven by a numeric score.

V. Lessons Learned

This paper reports several lessons learned through experiences applying specifications grading techniques. Allowing late submissions provides a mechanism for students to demonstrate competency when it is achieved, even if they are not able to do it before the original deadline. However, late submissions without consequences can undermine the positive benefits of deadlines and can even promote procrastination. Indeed, there is an opportunity cost imposed by late work, even if no grade penalty is assessed, and students should understand the consequences. In other words, despite the lack of penalty to grades, late work is not *free*.

While two-valued grading scales work, there can be benefits in using slightly richer grading scales that allow distinguishing between work that goes beyond the satisfactory expectations of an assignment, or distinguishing un-attempted work from work that is making progress towards meeting expectations.

Furthermore, introducing alternative grading approaches definitely requires a culture shift in the classroom, and instructors should approach it with this in mind. Without clear and repeated explanations of the justification for policy choices, the intended benefits, and the impact of student choices, students face anxiety and frustration with the introduction of such changes. Indeed, students see some potentially detrimental aspects of traditional points-based grading as tools they can use, and view alternative grading approaches as unfair because they can no longer "average out" under performance in required learning objectives. Finally, partially integrating alternative approaches with traditional points-based grading can be useful, but poses the risk of producing a strategy that has the worst disadvantages of both approaches, so hybrid grading approaches should be undertaken with appropriate thoughtfulness and caution.

With regard to educational tools, our two universities faced similar challenges when employing them within a specifications grading approach. This is not surprising, since the same learning management system and the same external learning tools were used at both institutions. It is clear that existing learning management systems and educational tools heavily rely on the assumption that grades are communicated and stored as numbers. Existing learning tools do not provide effective ways of grading using non-points-based schemes, or structuring feedback in a way that is not driven by a numeric score. Tools that grade in a monolithic percentage score are harder to employ in alternative grading approaches. A percentage grade does not differentiate what a student gets right and what they get wrong. A specification grade should indicate individually which of the specifications of the assignment have been met. Thus percentage scores hide the fact that some parts of the assignment might be incomplete.

In addition, the gradebook features of most online learning tools only support weighted averages in final grade calculations, making it difficult for students to understand what their grade will be when using approaches like Nilson's bundling described at UNC Charlotte.

Similarly, not all tools have the same support for late submissions. Some allow a due date and a separate "open until" date (i.e., late period) with optional penalties. This was implemented differently in all the tools we employed, making it a challenge to update dates. One tool simply blocked access to exercises after the last submission date. In the spirit of allowing students to continue practicing, tools should separate dates when they stop receiving submissions from dates used to calculate grades. Students should be able to practice even after the deadline when points are awarded. Some systems allow for this, but others do not.

Ideally, we want grading support systems to use multiple learning objectives per assignment, to represent student achievement in a manner that allows feedback on these different learning objectives to be teased apart, and to support richer options for resubmission attempts and late work. These apects of tool support are instrumental in supporting specifications grading and other alternative grading strategies.

VI. Conclusion

Alternative grading approaches are increasingly seeing interest among computing educators due to the many potential benefits they imply. In this experience report, we describe experiences adapting alternative grading practices such as specifications grading in introductory courses at two large universities. Through these experiences, we can confirm achieving some of the strengths of alternative grading schemes, including increased equity and greater flexibility for students to cope with and adapt to factors outside the classroom that can affect their performance in a course.

Nevertheless, the experiences reported here were positive and encourage us to continue exploring alternative grading practices and how to employ them. The most important lesson is that incorporating these practices is not simple or trivial and can pose significant challenges. However, through experience these approaches can be incorporated in ways that provide student benefits.

References

[1] Great Schools Partnership. (2020) Research supporting proficiency-based learning: Grading + reporting. [Online]. Available: https://www.greatschoolspartnership.org/proficiency-based-learning/research-evidence/research-supporting-ten-principles-grading-reporting/

[2] M. Townsley and T. Buckmiller. (2016) What does the research say about standards- based grading? a research primer. [Online]. Available: http://mctownsley.net/standards-based-grading-research/

[3] D. Reeves. (2008) Leading to change / effective grading practices. [Online]. Available: https://www.ascd.org/el/articles/effective-grading-practices

[4] J. Feldman, *Grading for equity: What it is, why it matters, and how it can transform schools and classrooms*. Corwin Press, 2019.

[5] W. J. Rapaport, "A triage theory of grading: The good, the bad, and the middling," *Teaching Philosophy*, vol. 34, no. 4, pp. 347–372, 2011.

[6] L. B. Nilson, *Specifications grading: Restoring rigor, motivating students, and saving faculty time*. Stylus Publishing, LLC, 2015.
[7] T. R. Guskey, "Grading policies that work against standards...and how to fix them," *NASSP Bulletin*, vol. 84, no. 620, pp. 20–29, 2000. [Online]. Available: https://doi.org/10.1177/019263650008462003
[8] Opendsa. [Online]. Available: https://opendsa-server.cs.vt.edu
[9] Codeworkout. [Online]. Available: https://codeworkout.cs.vt.edu
[10] S. H. Edwards, "Using software testing to move students from trial-and-error to reflection-in-action," in *Proceedings of the 35th SIGCSE Technical Symposium on Computer Science Education*, ser. SIGCSE '04. New York, NY, USA: Association for Computing Machinery, 2004, p. 26–30. [Online]. Available: https://doi.org/10.1145/971300.971312
[11] R. Talbert. Giving marks that indicate progress. [Online]. Available: https://gradingforgrowth.com/p/giving-marks-that-indicate-progress