Behind the Scenes of Delivering a Large Computing Course: The Experience of a TA Managing Logistics

Rachel S. Lim
University of California San Diego
La Jolla, CA, USA
ral077@ucsd.edu

Philip J. Guo
University of California San Diego
La Jolla, CA, USA
pg@ucsd.edu

Abstract

There are many tasks that must be done to keep a computing course running smoothly. In addition to pedagogical work, instructors do a multitude of behind-the-scenes administrative and organizational work themselves or delegate it to Teaching Assistants (TAs). Despite the pervasiveness of this type of work, the explicit details of what kinds of tasks are necessary, the processes to execute these tasks well, and the experiences and feelings of those doing the tasks have not been reported in the literature. Therefore, this experience report surfaces these details by presenting the reflections of a long-time TA who has worked on significant administrative tasks across eight CS courses at our institution over the years. As a representative sample of her experiences, we report on three anecdotes that highlight logistical and internal challenges she faced as a TA for a CS1 course: 1) managing the flow of 600 students coming into a computer lab to take proctored assessments, 2) scanning and uploading thousands of pages of on-paper exams, 3) leading an exam grading session of dozens of TAs. Based on her reflections, we see how much intentional foresight and preparation goes into preventing chaotic situations, as well as many concurrent and overlapping threads to manage in an overarching task. The combination of these factors as well as challenges in team and interpersonal dynamics were the primary sources of stress. Our goal is to bring awareness to this aspect of course delivery that is oftentimes hidden and inspire future research around it.

CCS Concepts

• **Social and professional topics → Computing education**.

Keywords

course administration; logistics; teaching assistants

ACM Reference Format:
Rachel S. Lim and Philip J. Guo. 2026. Behind the Scenes of Delivering a Large Computing Course: The Experience of a TA Managing Logistics. In *Proceedings of the 57th ACM Technical Symposium on Computer Science Education V.1 (SIGCSE TS 2026), February 18–21, 2026, St. Louis, MO, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3770762.3772547

1 Introduction

Computing courses are complex with many moving parts that must be coordinated together to deliver a pedagogically effective experience for students. For instance, a CS1 or CS2 course may include lectures, programming assignments, hand-graded written homework, on-paper exams, computer lab sessions, small-group discussion sections, and office hours. The content, timing, and logistics of all these parts must be carefully choreographed by the instructional staff, otherwise confusion and disorder may ensue.

For smaller courses, a single instructor often handles administrative work alone (in addition to the time-consuming work of teaching prep) since they might not have teaching assistants (TAs) to assist. For courses with higher enrollments, an instructor may organize a team of TAs to manage the complexities of scale.

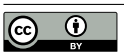
Yet despite the fact that instructors across many sizes of institutions all face the challenges of behind-the-scenes administrative work, *the computing education literature has not yet reported on what this invisible work consists of and what its unique demands are.* Instead, prior papers have emphasized the pedagogical aspects of course delivery or technical tools such as autograder software and I.T. infrastructure [2, 3, 18, 27, 31, 43]. This gap in coverage is notable because instructional staff may spend just as much (or even more) time on such administrative work as on teaching prep.

As a step towards filling this gap, this paper presents an experience report of an instructional staff member (the first author) who has managed administrative tasks across a wide variety of computing courses. The first author is a fourth-year graduate student (female, mid-20s) at a North American university who has been a TA for 20 quarter-long terms at the time of writing (as both an undergraduate and graduate student), eight in which she did significant behind-the-scenes work along with the professor.

To surface the challenges she tended to face, we present three anecdotes from a CS1 course that she was a TA for: 1) managing the flow of 600 students entering and exiting a computer lab to take in-person proctored assessments, 2) processing and scanning thousands of pages of on-paper exams to input into Gradescope [43] to grade, and 3) leading an exam grading session of dozens of TAs.

Upon critically reflecting on these experiences, we distilled higher-level themes. For instance, the first author, when coordinating tasks at scale, managed many threads at once and actively thought and planned ahead. This tended to cause stress for her both ahead of time and in time. More broadly, in cases where students and staff members are not aware of the possible logistical problems, it can be challenging yet important to communicate clearly to help them understand the importance of following protocols. Additionally, it can be difficult to find the right balance of structure, both with complexity and with empathy.

Our paper contributes a novel perspective to the computing education literature by foregrounding both 1) the range of complexities that may arise behind the scenes when managing a course, and



SIGCSE TS 2026, St. Louis, MO, USA.

ACM ISBN 979-8-4007-2256-1/2026/02
https://doi.org/10.1145/3770762.3772547

2) a firsthand account of how it was like to do this work and the challenges faced. Our goal for this work is twofold:

- To raise awareness of these hidden complexities and their unique demands
- To inspire future research on streamlining this process to reduce the burden on instructors and TAs.

Different instructors and TAs will have different experiences based on their contexts. We acknowledge the uniqueness of our experience, which is only one out of many, and we tell our story as an example to support our goals above.

2 Related Work

To our knowledge, our paper is the first to introspect on the unique challenges of doing the behind-the-scenes work to deliver a course. Many computing education papers focus on pedagogy or tools [2, 3, 18, 27, 31, 43], and we have found few that mention administrative work. Mirhosseini et al. surveyed 32 computing instructors and "identified several pain points, including [...] troubleshooting technical issues in students' computing environments, and endless administrative tasks" [34] but did not go into detail on these "endless administrative tasks". We enumerate examples of such tasks and give an in-depth account of one TA's experience doing them.

Although the course administration challenges we describe can emerge across scales, they become more complex as class sizes grow. Prior work on large enrollments addressed topics such as equitable grading at scale [4], managing large numbers of TAs [1, 13, 30, 41], improving autograder tools [18, 31, 43], I.T. infrastructure [27], scalable ways to provide individual feedback [44], and methods to increase students' sense of connection [22, 33]. However, there has not been as much work on overall logistical coordination. Some have described techniques to automate rote tasks [46], operate in-person testing centers [10], and keep course materials updated across different terms [20]. These prior studies focus on tools and techniques rather than presenting a personal viewpoint on the experience of coordinating logistical tasks, which our paper is the first to reveal.

In another thread of related work, lots of awareness has been raised about the pedagogical roles of TAs in computing courses, including interactions with students [5, 19, 28, 29, 37], grading and assessment [14, 38], and providing feedback [9, 48]. Markel and Guo report on a TA's struggles helping students with programming problems [29]. Riese et al. identified "becoming a professional TA" as a category of TA struggles, and it encompasses issues such as miscommunications between instructors and students, discovering unethical behavior of other TAs, and public speaking [40]. However, prior work has not yet described TAs' internal struggles with handling *non-pedagogical administrative* aspects of the job; our paper fills this gap in the literature.

The topics covered in TA training also reflect the aforementioned gap in prior work. TA training courses are offered in diverse forms [11, 16, 23, 24, 32, 35, 39]. However, the majority of topics covered are related to pedagogy, interactions with students, and grading. Huang and Fox also identified this common emphasis on "traditional" pedagogy in TA training programs, and in response they describe a climate-first curriculum that approaches pedagogical topics with a focus on classroom climate [16], while Muzny and Shah include more modules on DEI in their curriculum [35]. To our knowledge, there have been no publications about TA training courses that explicitly cover administrative tasks. Similarly, we have not found prior work on training instructors on these topics.

In fields outside of computing, there is work studying TAs' emotions [47], self-efficacy [6], self-image [12], and classroom management [25]. However, these past studies similarly focus on those topics in the context of pedagogy and classroom teaching, and they do not provide detailed firsthand accounts of TAs' experiences.

3 Context and Approach

Although instructors and TAs both perform behind-the-scenes work in computing courses, we present the perspective of a TA at a large North American university. We believe that a TA viewpoint is valuable for the community since TAs often see different types of courses and work with different instructors, so TAs are well-positioned to report from a range of contexts they have been in.

At the time of writing, the first author of this paper is a fourth-year graduate student (female, mid-20s) who has been a TA for 20 ten-week (academic quarter) terms as both an undergraduate and graduate TA. She has worked on a variety of courses including CS1, CS2, computer organization, CS theory, and compilers.

For this paper, we focus on one representative course that the first author TA'ed for in one term, a CS1 course that enrolled around 600 students. The staff consisted of 1 instructor, 5 graduate teaching assistants (GTAs), and 45 undergraduate teaching assistants (UTAs).

The instructor was responsible for lectures and higher-level curricular content. They delegated one major category of tasks for each of the five GTAs to take the lead on organizing: 1) programming assignments, 2) worksheets, 3) discussion sections, 4) exams, and 5) logistics. The instructor assigned these roles based on their prior collaborations with these five GTAs (either from other classes or research projects), taking into account their preferences and skills. The first author was responsible for "logistics", which was a broad role encompassing a variety of behind-the-scenes operations to ensure that all aspects of the course ran smoothly.

The instructor and all five GTAs had a sizable amount of work to do in their roles, and each of them likely faced different kinds of challenges that they could report. In this paper, our goal is to shed light on the administrative aspects of running computing courses and thus focus on the challenges faced by the first author.

We used a phenomenological approach for this study. Phenomenology is a qualitative method for research that investigates an individual lived experience from a single first-person perspective, uncovering insights through deep and detailed reflection from the person [17]. Rather than seeking generalizable findings, it prioritizes depth and nuance in the understanding of a single person's experience.

During the term described in this report, the first author took field notes about her observations, experiences, and feelings as a TA for the course. Each week, she met with the second author to discuss and reflect on these events. The second author is a professor from another department who was *not* involved in any of the 20 courses that the first author TA'ed for; they have a background in ethnographic and phenomenological research in computing education, so they helped the first author to critically examine and distill her thoughts. After the term ended, we first examined the

notes from a higher level and extracted common threads among the challenges that the first author faced. We then identified specific representative anecdotes. From there, we noticed different layers of challenges, so we structured those into an outline that we continued to change and iteratively improve as we wrote, discussed, and saw how the story was coming together.

We took a similar approach as prior SIGCSE experience reports that are based on one author's firsthand experiences [21, 26, 29], so the usual limitations about lack of generalizability apply too.

4 Experiences and Reflections on Course Administration Tasks

For this section, we switch to using the first-person voice to report the first author's personal experiences and feelings. Specifically, we focus on the challenges she faced with behind-the-scenes administrative tasks in the context of this CS1 course. Because the specific details and circumstances for those at other institutions may not match those of ours, our goal is not to generalize this instance to other contexts. Our experience is situated on only one point on a spectrum of diverse experiences that instructors and TAs could have. We share what we learned from our end so that the community can draw broader insights from it and discuss their opinions from their own perspectives and contexts.

4.1 Three Representative Example Challenges of Managing Course Logistics

I begin this section by presenting three examples of behind-the-scenes tasks that were logistically complex, and despite being gratifying, they also caused stress for me. Although these all come from a single CS1 course, they are broadly representative of the types of tasks that the instructor, other TAs, and I performed in other computing courses that I had TA'ed for in other terms.

1. Overseeing an all-day exam event

Task: Two times during the term, our course held mandatory coding assessments in the computer labs. These were graded assessments where students had to solve a coding problem in an in-person proctored environment. We needed to plan for and coordinate 600 undergraduate students coming in and out of four different rooms every hour for eight consecutive hours to take their exams. We asked students to come in during their regularly-scheduled section times, so following section enrollment, each hour we had between 50 and 100 students (of the 600 total) scheduled to come into one of the four lab rooms.

Execution: Although it did not matter which of the four rooms the students went to, I felt that having up to 100 people roaming the hallways trying to get to "any of these rooms" without specific instructions on where to go would cause confusion and create an overwhelming situation for both students and staff. Thus, I sent a list of room assignments to students ahead of time. I also predesignated a UTA to stand in the hallway with that list in case students were unsure of where to go.

Additionally, we needed to reset the room in between sessions (e.g., make sure all lab computers were logged out), and I knew that if we did not communicate clearly to all the students to wait outside until we were ready, some would wander in prematurely. We preempted this issue by posting signs on the doors asking students to wait and by designating another UTA to stand at the entrance of each room to ask any students who still happened to come in to go back outside. Waiting to let a large group of students in at the same time also reduced the number of times we needed to repeat any verbal instructions that we wanted to give students upon entry.

We did not assign students specific seats within the lab rooms, but we did want them to spread out and not sit directly adjacent to each other. To prevent the stress of having dozens of nervous students figuring that out with each other on the spot, we told students to sit in any evenly-numbered seat in the room. We communicated this several times (ahead of time in class, added as instructions on room assignment list, written on the whiteboard of the room, verbally announced when students entered) to increase the chances of students following the instructions. We also designated a UTA to monitor this process and help students find seats if needed.

Reflection: To my relief, the entire day ran smoothly. Despite my nervousness, I anticipated almost all potential issues and implemented procedures to either prevent them or to resolve them quickly. However, this outward appearance of calm contrasted with my anxious internal mental state both before and throughout the all-day eight-hour event. Reflecting back on this now, I feel that what made things run smoothly was my *vigilance and proactive thinking to prevent chaotic situations from arising.*

My preparation consisted of first identifying a possible issue and then figuring out what actions the staff could take to prevent it. For example, I anticipated that if a student came in to the lab too early and we did not address it quickly, too many instances of that could escalate to the rest of the students also coming in because they see others inside already, and I would not be able to act fast enough to address all these cases on my own. I had experienced that before myself when I was a TA for a prior class in which we had hosted coding exams in the same lab rooms. We could prevent this situation by stopping students from entering early, hence placing signs on doors. I also knew that despite our pre-intervention, some students would still not follow directions or not see the signs, so as a backup, I had a UTA stand near the entrance to regulate traffic.

This level of logistical pre-planning needed to be repeated for every problematic situation I could think of. Thus, I was stressed thinking about multiple different threads of possible issues – for each one, coming up with solutions, communicating with fellow staff members, and taking immediate actions – all while considering how each thread might impact the others. With all of that, I needed to make sure that my plans were not overly-complicated as to slow things down too much, confuse people, or cause other problems. And I was continually stressed that either my plans would go awry or that I had missed some other unexpected contingency.

Reflecting more broadly beyond this specific example, in general it is stressful for me to think about what could go wrong ahead of time because I am putting my mind into a 'paranoid' headspace, which makes me feel the stress of those hypothetical worst-case scenarios. *I essentially need to create this kind of pre-stress for myself ahead of time in order to figure out how to prevent more stress later for myself, dozens of course staff members, and hundreds of students.*

Broader relevance: The challenges and stress I encountered are not limited to this specific situation. In large classes, other logistical

puzzles such as how to schedule each student for a meeting with a staff member or how to efficiently get all physical exam papers back to each student after grading may require similar levels of pre-planning and coordination. Even in classes that are not as large, issues such as limited physical space and not enough TAs may pose constraints that necessitate similar levels of planning.

2. Scanning hundreds of paper exams

Task: We use Gradescope [43] to electronically grade exams. Before we could do any grading, we first needed to scan all the physical exam papers to upload to Gradescope, and my role was to coordinate the process of scanning and uploading.[1] Each of the 600 students' exam packets consisted of nine pages, each double-sided, and there were three different versions (to mitigate cheating). It was very important for students' sake to not lose or mix up any of the pages.

Execution: All 600 packets were counted, numbered, divided into stacks, and stapled corners cut off. Then stacks went through the scanner and were all kept track of individually until the scan was confirmed. Finally, the PDF files from the scans were received, confirmed, renamed systematically, and uploaded to Gradescope. I coordinated this process by having a few UTAs work in parallel.

Several other TAs who had not been tasked with scanning voluntarily came by to offer more help. Even though this was a kind offer, the nature of the task required a level of coordination and precision that made it challenging to integrate additional support. For example, we were writing a unique identification number on each exam packet, and this task would have been difficult to split up and parallelize due to the need for coordination. We were also limited by physical resources and space – we only had access to one paper cutter. Additionally, the printing area was not large enough to spread everything out like we wanted to, so adding more people to the process would have made our tight space even more crowded. Thus, we thanked them for their offer and continued on without them. The process went smoothly, and we finished in less time than the instructor expected.

Reflection: I had to manage and keep track of many things as they were all happening simultaneously. New exam packets were arriving in waves from the exam rooms. I had to designate a specific space to put exams at each point in the process (e.g., unprocessed, numbered, paper corners cut off) and make sure each UTA knew exactly which areas to take from and to deposit into based on the task they were working on. We processed one of the three exam versions at a time, and I had to monitor the transition from one version to the next. There is a several-minute delay from when the papers go through the scanner to when the scanned PDF file becomes visible in my inbox to be confirmed and uploaded, so I had to context-switch to do other tasks in the meantime while also always keeping an eye on my inbox. Managing all of this at the same time pulled me in all kinds of directions, and it was stressful to have so much on my mind at once.

This process was also delicate and prone to error. For example, any TA (perhaps in a rush) could accidentally bump the table and get all the unstapled papers mixed up, which could be very hard to recover from, especially in this instance when not all students wrote their names on all nine pages. Or a TA could miscount the number of exams in their stack at the beginning, and that would create extra work later to figure out where the mistake originated. Or the piles could get put in wrong places and we could lose track of which ones were already scanned. Knowing that these problems could be caused and arise at any point added to the stress and made me feel on-edge the entire time.

Broader relevance: I was fortunate to have support from other TAs to accommodate for the massive volume of paper to scan and keep track of. My challenge was in coordinating and keeping track of all the concurrent tasks, even though I had people helping me to do them. Similar types of challenges and stress can arise in other contexts as well. For example, an instructor of a small class may scan all of the exams on their own, and although they do not need to manage other people, their process could still require them to keep track of too many threads concurrently. Zooming out from only exam-scanning, any high-stakes situation that requires a lot of coordination can lead to similar levels of fear and nervousness that something might go wrong.

3. Managing an exam grading session

Task: After all paper packets were scanned into Gradescope, we held a three-hour staff meeting to grade the midterm exam. Each GTA was assigned to supervise the grading of one free-response question with a team of around eight UTAs.

Execution: Due to factors such as my group arbitrarily getting the longest question (worth the most points and involving more open code-writing) and me taking longer upfront to come up with a robust grading rubric, other groups were long done grading their questions by the time my group was only halfway done. At that point, two other groups of nine staff offered to help my group finish grading our question. I accepted their offer even though my intuition was that adding that many people would complicate the process.

When they joined, I paused my own grading to explain the rubric for our question to all the new graders and demonstrate grading a few submissions in front of them. Since the 18 new graders wanted to start right away, I was not able to assign them specific submissions to grade. Multiple graders used the "next ungraded submission" button in Gradescope to find a submission to grade, but this created a race condition [36] where multiple people were trying to grade the same submission at the same time. Additionally, a UTA from my group noticed after the meeting that some of the new graders had applied rubric items incorrectly, so we had to identify and re-grade those submissions.

Reflection: I was continually stressed because I knew, even before the new graders started grading, what complexities this situation would introduce. I knew that when the new graders first started grading, it would take time for them to become familiar with the exam question and establish good context, and I foresaw that this would cause the potential for more mistakes in grading as well as a large volume of clarifying questions from them that I might struggle to handle in time. I also knew that not assigning specific submissions to each person would cause confusion and disarray.

Though I was very aware of what would happen, it was difficult to communicate that in the moment for others to understand,

[1]The TA who was in charge of exams focused on creating content for the exams, while I led the scanning because it involved coordinating staff and a process.

particularly as the graders were eager to get started and I felt a lot of pressure. The workflow strategies that would have helped us complete these tasks more efficiently (e.g., having fewer new graders join, taking extra time to assign specific submissions) may also seem counterintuitive, and that adds additional steps in helping others to see and grasp the value in these processes. I myself, even after many years of knowing and experiencing what would be the most optimal, still sometimes feel a bit of an instinctual resistance to taking a course of action that seems like it could be incorrect.

Broader relevance: With any amount of people working together, similar challenges can arise in figuring out and implementing the most effective process. This can be especially true when the chosen method requires more work to be done earlier in the process and the benefit of that additional work cannot be seen until later.

4.2 Recurring Higher-Level Themes that Arose Across Multiple Examples

4.2.1 Others may not be aware of these challenges. In all of the examples described above, the ability to execute these tasks smoothly relied on a foundation of prior experience and knowledge. With the in-lab coding assessments, I knew what to prepare for and anticipate because I had organized these events in the past. Similarly, I had scanned exams in many previous classes, and I had mixed up exam papers myself. From that, I learned how tedious and stressful it is to correct such an error, and I also learned what can be done earlier in the process to avoid this kind of trouble. I had not previously been in a situation specifically similar to the exam grading one, but I was able to draw from my general experience in managing teams of people to have an intuition of what would have optimized our workflow.

In this particular class, the majority of the staff were in their first term working as TAs in our department, and generally in past classes I have seen the majority of the staff have three quarters or less of prior TA experience, therefore they have not yet had as much time to learn about course administration logistics and gain relevant experiences. Additionally, if I or someone else handles a situation smoothly (e.g., the computer lab and exam scanning situations), others will not get to witness things going wrong and experience what could have happened otherwise.

One resulting challenge is that others may not understand why I want to implement certain measures at the beginning. They may not agree with my plans because to them, the additional work created at the beginning of the processes does not seem to be worth the trouble, and that is understandable if they are not aware of the problems down the line that I am trying to prevent. For instance, when we were grading exams at staff meeting and I wanted to spend more time upfront assigning specific submissions to people to grade, some TAs were opposed to this because without knowing the consequences of not doing so, it would be reasonable to deduce that starting the grading sooner would help us finish faster. Especially in cases like this where my plans may seem counterintuitive at first glance, I have the added responsibility of trying to help others understand my reasoning for having them do certain tasks.

Reflecting more deeply, me knowing that most staff members do not realize the extent to which preparation work is necessary makes me feel lonely and isolated, since I feel like I am in some ways alone in foreseeing what might go wrong and in doing a lot of invisible work. I feel alone in bearing the burden of planning everything correctly and predicting everything that may go wrong, and much of it is self-inflicted pressure. Though I enjoy this kind of logistics work and planning, the job can feel especially hard when the extent of the challenges I face on the job is mostly unknown to others.

4.2.2 Balancing structure and empathy. Section 4.1 showed the amount of complexity there can be in managing a situation. The more intertwined pieces there are, the more potential there will be for small deviations to snowball into big inconveniences. Thus, I find that the more people there are to coordinate, whether students or staff, the more important it is that everyone follows instructions and completes things in a timely manner. This gets hard when TAs or students need to ask for exceptions such as a deadline extension on a task or an assignment, or if someone wants me to make a change to something that has already been set. It is not hard to manage one extension or to make one change, but when there is potential for too many requests, I have to end up being very firm about not allowing for exceptions, barring extenuating circumstances.

However, my struggle here is that people might see this as being overly strict or even mean. To them, it might seem insignificant to ask for a small exception, as most people do not get the chance to see the bigger picture of how their request affects the course more broadly. Even though I do not find it difficult to be strict to enforce the policies that I set, I do find that when saying 'No' to someone who requests an exception, it helps for me to explain the higher-level impact of their request. When the requester understands the situation better, they often find my decision more reasonable, and it leaves both them and me more at ease.

Another challenge here is that at the same time, I want to be kind and understanding of people and the different situations they might be in. I want to be accommodating and extend grace as much as I can. There are circumstances in which I would be happy to help someone out by making an exception for them. However, it gets difficult when that is opposed by the increasing potential for an overwhelming amount of disorganization and confusion, especially in a large class. Finding the right balance between empathy for people and steadfastness for process is an ongoing challenge I still struggle with, and I believe other instructors and TAs may as well.

4.2.3 Communication and expectation. Clear communication is required as a first step to direct everyone to complete their assigned tasks correctly. It is important to give instructions that unambiguously outline the task and provide all the information needed to complete it, thus reducing confusion and making it easier for everyone. This is a challenge since I need to anticipate upfront how my instructions may be unclear or misinterpreted.

The other half of the equation is that the students and TAs receiving the instructions need to understand, interpret, and execute them. Younger undergraduate students may not yet have learned the importance of reading and following instructions carefully. Similarly, some newer UTAs may have had limited exposure to professional settings, therefore they may not yet fully grasp the basic standards of responsibility that this job (and any other job) would require, such as checking notifications and completing tasks in a timely manner. In some previous courses that I have TA-ed

for, we took time at the start of the term to explicitly communicate expectations to the entire TA staff; that way everyone, not just the newer TAs, was made aware of all their responsibilities. I have noticed in these quarters that the staff runs a lot more smoothly.

5 Discussion

5.1 Should Logistics Be a First-Class Concern?

For instructors and TAs alike, teaching is a big focus of the job, as it should be. It is very important and necessary to actively think about our teaching practices, content delivery, the student experience, and how to improve all of those. However, the emphasis on teaching often causes the logistical aspects to be overlooked. Many instructors and TAs may find administrative and management tasks less exciting and more burdensome, as many are drawn to this job primarily because they want to work with students.

People have different reasons for choosing to work in this role, but it is unlikely for someone's main motivation to be a passion for logistics. However, the reality is that courses cannot run without logistics, and in combination with low enthusiasm, it is even more necessary to be intentional about developing skills to do behind-the-scenes tasks effectively and efficiently. Refining and improving on our teaching is always a first-class concern, but alongside that, we must remember that investing in good logistics and management upfront is ultimately what will create the space in the future for all of the exciting pedagogical parts of this job to flourish.

5.2 How Do We Pass On Logistical Knowledge?

Much knowledge about how to do this job well is discovered through personal experience. This knowledge needs to be passed down intentionally because TAs who may gain this knowledge over time will eventually finish their degrees and graduate. If the knowledge is not passed down, every generation of TAs will go into these logistical tasks unprepared and re-discover the same problems.

However, it can be hard to document all the nuances and knowledge of the experience in writing (i.e., it is a form of *tacit knowledge* [7]). In section 4, we described only a fraction of what the first author has experienced. Additionally, it would be difficult to capture in words something like how to react to an unexpected situation, as that is a skill that is developed through experience.

We believe approaches like *cognitive apprenticeship* [8, 42] are promising: an experienced TA gets paired with a new TA, and they work together on some representative tasks. This allows the experienced TA to walk the new TA through their thought process and show them the nuances, and it enables the new TA to directly witness and be a part of the process, which is a richer experience than reading about others doing it or simply being told what to do.

Zooming out, perhaps the biggest challenge here is that even if we could figure out a good way to pass down such tacit knowledge and experience, how would we even know *what* we should pass on? It may be difficult to identify which of the many pieces of knowledge about TAing a specific course would be significant enough to share, and we cannot precisely predict what kinds of logistical situations will unfold in future terms. Additionally, much of what we know from experience may seem trivial after a while, and thus we may forget that it could be helpful to explain to others (sometimes called the *curse of knowledge* [15, 45]). The open question we pose for future researchers is: How can TAs pass down what is most critical without being overwhelmed by too many details?

5.3 What Would Have Reduced the Stress?

We showed that for the first author, logistical work, even when done well, carried a lot of emotional weight, and trying to spread it out among more people did not relieve that weight. Because some of that difficulty stemmed from others not being aware of what it takes for a course to run smoothly and how everyone plays an important part of the bigger picture, a first step towards improvement could involve teaching this to instructors and TAs as part of their training. This may lead staff to be more mindful of their actions and decisions, which will ultimately make it easier for the person coordinating.

Future work could explore whether there are other possible methods of work that similarly maintain order in courses but without as many factors that may cause stress. For example, if the person managing logistics were to fully delegate the management of each thread of a complex task to a different person, how would the challenges change? Is there a point where the costs of maintaining impeccable order reach diminishing returns?

Expanding outwards from just those who manage logistics, any instructor or TA who works on any aspect of course delivery can face challenges in their role. More conversations should be raised about how this job affects the people doing it, and that will hopefully humanize the role and create more space for instructional staff to express how they are feeling and what they need. If we normalize checking in on each other and come together to support each other as people, then the stress that we experience in these roles might feel a little less heavy and a little less isolating.

6 Conclusion

This paper presented the experience of the first author, who managed logistics as a TA for a large CS1 course and many courses before that. The first author has been a TA at the same institution for almost every term over the past seven years and has thought extensively about the different aspects of course delivery as a computing education researcher for four of those years. This enabled her to gather these higher-level insights after learning, observing, and being a part of many courses and staffs through the years.

Limitations and Future Work: We reported only on a single TA's experiences in a course with many staff members, each of whom were responsible for different aspects. Thus, the instructor and other TAs likely all faced their own challenges and have unique stories to tell. Furthermore, TAs at different institutions may face unique challenges due to differences in local context.

Our paper follows in the tradition of SIGCSE experience reports based on one author's firsthand experiences [21, 26, 29]. We encourage the computing education community to surface more firsthand narratives to shed light on otherwise-hidden aspects of running a course. Doing so will hopefully inspire more experimental and tools research into ways to improve this process.

Acknowledgments

We thank the anonymous reviewers for their comments and suggestions, as well as Joe Politz, who provided helpful feedback.

References

[1] Ildar Akhmetov, Sadaf Ahmed, and Kezziah Ayuno. 2024. How We Manage an Army of Teaching Assistants: Experience Report on Scaling a CS1 Course. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education.*

[2] Brett A. Becker and Keith Quille. 2019. 50 Years of CS1 at SIGCSE: A Review of the Evolution of Introductory Programming Education Research. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education.*

[3] Noelle Brown, Benjamin Xie, Ella Sarder, Casey Fiesler, and Eliane S. Wiese. 2024. Teaching Ethics in Computing: A Systematic Literature Review of ACM Computer Science Education Publications. *ACM Trans. Comput. Educ.* 24, 1, Article 6 (Jan. 2024), 36 pages.

[4] A. T. Chamillard and Laurence D. Merkle. 2002. Management challenges in a large introductory computer science course. In *Proceedings of the 33rd SIGCSE Technical Symposium on Computer Science Education (SIGCSE '02).*

[5] Alan Y. Cheng, Ellie Tanimura, Joseph Tey, Andrew C. Wu, and Emma Brunskill. 2024. Brief, Just-in-Time Teaching Tips to Support Computer Science Tutors. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1 (SIGCSE 2024).*

[6] Pit Ho Patrio Chiu and Paul Corrigan. 2019. A study of graduate teaching assistants' self-efficacy in teaching: Fits and starts in the first triennium of teaching. *Cogent Education* 6, 1 (2019).

[7] Anna T. Cianciolo and Robert J. Sternberg. 2018. *Practical Intelligence and Tacit Knowledge: An Ecological View of Expertise* (2 ed.). Cambridge University Press, 770–792.

[8] Allan Collins, John Seely Brown, and Ann Holum. 1991. Cognitive apprenticeship: Making thinking visible. *American Educator* 15, 3 (1991), 6–11.

[9] Amy Cook, Vinhthuy Phan, and Alistair Windsor. 2022. Improving TA Feedback on In-Class Coding Assignments for Introductory Computer Science. In *Proceedings of the 27th ACM Conference on Innovation and Technology in Computer Science Education Vol. 1 (ITiCSE '22).*

[10] Kelly Downey, Kris Miller, Mariana Silva, and Craig Zilles. 2024. One Solution to Addressing Assessment Logistical Problems: An Experience Setting Up and Operating an In-person Testing Center. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1 (SIGCSE 2024).*

[11] Francisco J. Estrada and Anya Tafliovich. 2017. Bridging the Gap Between Desired and Actual Qualifications of Teaching Assistants: An Experience Report. In *Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '17).*

[12] A Flaherty, A O'Dwyer, P Mannix-McNamara, and JJ Leahy. 2017. The influence of psychological empowerment on the enhancement of chemistry laboratory demonstrators' perceived teaching self-image and behaviours as graduate teaching assistants. *Chemistry Education Research and Practice* 18, 4 (2017), 710–736.

[13] Jeffrey Forbes, David J. Malan, Heather Pon-Barry, Stuart Reges, and Mehran Sahami. 2017. Scaling Introductory Courses Using Undergraduate Teaching Assistants. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education (SIGCSE '17).*

[14] Pontus Haglund, Linda Mannila, Filip Strömbäck, and Aseel Berglund. 2024. Grasping the Unseen: TA Insights into Teaching Subtle Concepts in Computer Science. In *Proceedings of the 2024 on Innovation and Technology in Computer Science Education V. 1 (ITiCSE 2024).*

[15] Chip Heath and Dan Heath. 2006. The curse of knowledge. *Harvard Business Review* 84, 12 (2006), 20–23.

[16] Victor Huang and Armando Fox. 2023. A Climate-First Approach to Training Student Teaching Assistants. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1 (SIGCSE 2023).*

[17] Dawn Joseph. 2014. Interpretative phenomenological analysis. *Research methodologies in music education* 7, 4 (2014), 145–165.

[18] Hieke Keuning, Johan Jeuring, and Bastiaan Heeren. 2018. A Systematic Literature Review of Automated Feedback Generation for Programming Exercises. *ACM Trans. Comput. Educ.* 19, 1, Article 3 (Sept. 2018), 43 pages.

[19] Sophia Krause-Levy, Rachel S. Lim, Ismael Villegas Molina, Yingjun Cao, and Leo Porter. 2022. An Exploration of Student-Tutor Interactions in Computing. In *Proceedings of the 27th ACM Conference on Innovation and Technology in Computer Science Education Vol. 1 (ITiCSE '22).*

[20] Sam Lau, Justin Eldridge, Shannon Ellis, Aaron Fraenkel, Marina Langlois, Suraj Rampure, Janine Tiefenbruck, and Philip J. Guo. 2022. The Challenges of Evolving Technical Courses at Scale: Four Case Studies of Updating Large Data Science Courses. In *Proceedings of the Ninth ACM Conference on Learning @ Scale.*

[21] Colleen M. Lewis. 2021. Physical Java Memory Models: A Notional Machine. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education (SIGCSE '21).*

[22] Miranda Li, Ali Malik, and Chris Piech. 2025. Fostering and Understanding Diverse Interpersonal Connections in a Massive Online CS1 Course *(SIGCSE 2025).*

[23] Runda Liu, Shengqi Chen, Jiajie Chen, Songjie Niu, Yuchun Ma, and Xiaofeng Tang. 2025. Iterative Design of a Teaching Assistant Training Program in Computer Science Using the Agile Method. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 1 (SIGCSETS 2025).*

[24] Madeleine Lorås. 2020. From teaching assistants to learning assistants–lessons learned from learning assistant training at Excited. *Læring om læring* 4, 1 (2020).

[25] Jiali Luo, Laurie Bellows, and Marilyn Grady. 2000. Classroom management issues for teaching assistants. *Research in Higher Education* 41 (2000), 353–383.

[26] David J. Malan. 2010. Reinventing CS50. In *Proceedings of the 41st ACM Technical Symposium on Computer Science Education (SIGCSE '10).*

[27] David J. Malan. 2024. Containerizing CS50: Standardizing Students' Programming Environments *(ITiCSE 2024).*

[28] Yana Malysheva, John Allen, and Caitlin Kelleher. 2022. How Do Teaching Assistants Teach? Characterizing the Interactions Between Students and TAs in a Computer Science Course. In *2022 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC).* 1–9.

[29] Julia M. Markel and Philip J. Guo. 2021. Inside the Mind of a CS Undergraduate TA: A Firsthand Account of Undergraduate Peer Tutoring in Computer Labs. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education.*

[30] Emma McDonald, Gisele Arevalo, Sadaf Ahmed, Ildar Akhmetov, and Carrie Demmans Epp. 2023. Managing TAs at Scale: Investigating the Experiences of Teaching Assistants in Introductory Computer Science. In *Proceedings of the Tenth ACM Conference on Learning @ Scale (L@S '23).*

[31] Marcus Messer, Neil C. C. Brown, Michael Kölling, and Miaojing Shi. 2024. Automated Grading and Feedback Tools for Programming Education: A Systematic Review. *ACM Trans. Comput. Educ.* 24, 1, Article 10 (Feb. 2024), 43 pages.

[32] Mia Minnes. 2022. Designing TA Training for Computer Science Graduate Students: Remote and Self-paced Options for A Supported Introduction to Reflective Teaching. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education - Volume 1 (SIGCSE 2022).*

[33] Mia Minnes, Christine Alvarado, and Leo Porter. 2018. Lightweight Techniques to Support Students in Large Classes. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education (SIGCSE '18).*

[34] Samim Mirhosseini, Austin Z. Henley, and Chris Parnin. 2023. What Is Your Biggest Pain Point? An Investigation of CS Instructor Obstacles, Workarounds, and Desires. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1 (SIGCSE 2023).*

[35] Felix Muzny and Michael D. Shah. 2023. Teaching Assistant Training: An Adjustable Curriculum for Computing Disciplines. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1 (SIGCSE 2023).*

[36] Robert HB Netzer and Barton P Miller. 1992. What are race conditions? Some issues and formalizations. *ACM Letters on Programming Languages and Systems (LOPLAS)* 1, 1 (1992), 74–88.

[37] Leah Perlmutter, Jean Salac, and Amy J. Ko. 2023. "A field where you will be accepted": Belonging in student and TA interactions in post-secondary CS education. In *Proceedings of the 2023 ACM Conference on International Computing Education Research - Volume 1 (ICER '23).*

[38] Emma Riese. 2022. *Perspectives on assessment in introductory computer science courses: Exploring and comparing experiences of students, teaching assistants, and course coordinators.* Ph. D. Dissertation. KTH Royal Institute of Technology.

[39] Emma Riese and Viggo Kann. 2022. Training Teaching Assistants by Offering an Introductory Course. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education - Volume 1 (SIGCSE 2022).*

[40] Emma Riese, Madeleine Lorås, Martin Ukrop, and Tomáš Effenberger. 2021. Challenges Faced by Teaching Assistants in Computer Science Education Across Europe. In *Proceedings of the 26th ACM Conference on Innovation and Technology in Computer Science Education V. 1 (ITiCSE '21).*

[41] Eric S. Roberts. 2011. Meeting the challenges of rising enrollments. *ACM Inroads* 2, 3 (aug 2011), 4–6.

[42] Anshul Shah and Adalbert Gerald Soosai Raj. 2024. A Review of Cognitive Apprenticeship Methods in Computing Education Research. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1.*

[43] Arjun Singh, Sergey Karayev, Kevin Gutowski, and Pieter Abbeel. 2017. Gradescope: A Fast, Flexible, and Fair System for Scalable Assessment of Handwritten Work. In *Proceedings of the Fourth (2017) ACM Conference on Learning @ Scale (L@S '17).*

[44] Sven Strickroth. 2024. Scalable Feedback for Student Live Coding in Large Courses Using Automatic Error Grouping. In *Proceedings of the 2024 on Innovation and Technology in Computer Science Education V. 1 (ITiCSE 2024).*

[45] Carl Wieman. 2007. The "curse of knowledge" or why intuition about teaching often fails. *American Physical Society News* 16, 10 (2007), Q5.

[46] Chris Wilcox. 2015. The Role of Automation in Undergraduate Computer Science Education. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE '15).*

[47] Lan Yang, Chia-Ling Hsu, Tianfang Ye, and Kuen Fung Sin. 2022. Assessing emotions of teaching assistants in inclusive education. *Frontiers in psychology* 13 (2022).

[48] Alina Zaman, Amy Cook, Vinhthuy Phan, and Alistair Windsor. 2023. A Practical Strategy for Training Graduate CS Teaching Assistants to Provide Effective Feedback. In *Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 1 (ITiCSE 2023).*