Leveraging Senior Undergraduate Teaching Assistants in Computer Science Education: A Case Study from a Swedish University

Kamilla Klonowska[1], Daniel Einarson[2]

Kristianstad University, Sweden[1,2]

Abstract

As Computer Science (CS) programs continue to grow in enrollment and complexity, even small universities face increasing demands on teaching resources, student support, and course quality. One widely adopted strategy to address these challenges, especially in large institutions, is the employment of Teaching Assistants (TAs), particularly undergraduate TAs (UTAs). However, the role and effectiveness of TAs in smaller university settings, where resources are more limited and faculty are more tightly stretched, remain underexplored. This study investigates the triple benefits, or rather "win-win" outcomes, of employing TAs in CS education within a small university context. From the instructors' perspective, TAs can significantly reduce workload by assisting with grading, answering student questions, facilitating lab sessions, and improving student engagement. Simultaneously, TAs gain valuable pedagogical experience, deepen their subject knowledge, and develop transferable skills in communication and leadership. And lastly, from the students' perspective, TAs serve not only as supportive instructors but also as relatable peers who can offer valuable insights into study strategies, exam preparation, effective communication with faculty, and navigating the overall academic experience.

Keywords: *"Undergraduate Teaching Assistants (UTAs)", "Computer Science Education", "Peer-to-Peer Learning", "Student Engagement and Communication"*

1. Introduction

In large universities, Teaching Assistants´ (TAs) roles in computer science education are typically filled by PhD students or Graduate Teaching Assistants (GTAs). However, in smaller universities where PhD students are not available, an alternative approach is to engage senior undergraduate students as teaching assistants, often referred to as Undergraduate Teaching Assistants (UTAs).

A pioneering initiative of the use of UTAs in computer science education at Stanford University, 1988, is presented in [1], which addresses a strategic response to three pressing challenges: (1) increasing student enrollment, (2) improving the quality of education, and (3) maintaining costs at existing levels. Reflecting on their model, several key elements stand out as timeless and remarkably forward-thinking. First, the decision to use undergraduates rather than graduate students was not only cost-effective but based on the belief that UTAs could often be more effective peer educators, i.e. closer in experience to the students and better able to relate to their challenges. Second, the program emphasized the importance of organization and communication, with undergraduates themselves coordinating logistics and holding weekly meetings to align programming assignments across introductory courses. This created a strong sense of community and consistency across the curriculum. The teaching philosophy behind the program was the third key. It was rooted in the idea that excellent teaching is not just about deep subject knowledge; it is about listening, engaging, and continuously improving one's pedagogical skills. The Stanford team saw teaching as a craft that evolves with practice and reflection. The importance of building strong relationships between section leaders and students was another cornerstone of their approach. Weekly discussion sections, interactive grading, and small group sizes allowed for individualized support and meaningful feedback. Finally, the emphasis on continuous training and feedback was critical. Weekly meetings focused on teaching techniques, supported by peer mentoring and the sharing of experiences among section leaders. This commitment to reflective teaching practice empowered UTAs to grow in confidence and competence. In many ways, this early model anticipated many of the best practices we associate with effective peer teaching today. It reminds us that with the right structure, support, and philosophy, UTAs can play a transformative role in computer science education.

Building on this foundation, Mirza et al. [2] present a comprehensive systematic literature review examining the practices and claimed benefits of UTA programs in computer science education. They approach the topic from two perspectives: first, by developing a taxonomy of practices related to the

design and implementation of UTA programs; and second, by identifying and evaluating the reported benefits of such programs, including the strength of the supporting evidence. The study identifies five key themes in UTA program practices: recruiting and selection, duties, training, organizational structure, and program evaluation. A variety of UTA roles are described in the literature, including lab assistance, grading, peer instruction, and even lecture delivery. However, recruitment criteria and program structure vary widely between institutions. Training practices range from formal coursework to peer mentoring and self-reflection.
In terms of benefits, the review [2] categorizes reported advantages for UTAs, students, instructors, and institutions.
For UTAs, common benefits include improved technical and interpersonal skills, increased confidence, and greater engagement with the subject.
For students, UTAs enhance classroom support, offer relatable peer guidance, and contribute to improved learning outcomes and satisfaction.
Instructors benefit from reduced workloads and enhanced student feedback, while
institutions see gains in community building and cost-effectiveness.
A key finding, however, is that most of the reported benefits remain anecdotal, with relatively few supported by rigorous, empirical research. The authors emphasize the need for more structured studies linking specific practices to measurable outcomes and call for a common reporting framework for UTA program research. This review offers valuable insights for educators looking to establish, expand, or improve UTA programs, while also highlighting critical research gaps—particularly in the areas of diversity, training efficacy, and long-term impact on participants.
The UTA model creates a win-win situation by fostering a more approachable learning environment, where younger students feel more comfortable asking questions and seeking guidance from their more experienced peers. Student engagement plays a crucial role in the success of this approach, as peer-to-peer interactions differ from traditional student-teacher relationships, a distinction also noted by the authors in [3]. Various communication channels exist between students and teachers, as well as among students themselves.
This article presents a UTA model implemented in two courses within a Bachelor´s Programme in Computer Science, targeting first- and second-year students. The communication between students and UTAs is supported by the use of Discord (software), that provides a flexible and informal platform for students to reach out to the UTAs even during evenings and weekends. It is worth mentioning that teachers do not participate on Discord, thus being relieved from time-consuming but still important efforts. Involving older students as UTAs enhances their own learning experience, strengthens their CVs, and equips them with valuable teaching and mentoring skills for their future careers. Through experimenting with different models, this article discusses the challenges, benefits, and practical insights gained from implementing this model in a small university setting. Furthermore, it presents feedback from students, teachers and UTAs to evaluate the effectiveness of this approach.

2. Method

This research is based on data collected from students' course evaluations during last 3-5 years in two courses targeting first- and second-year students, where second- and third-year students served as teaching assistants. In addition, data was gathered through interviews with the TAs and reflections from courses´ instructors. Both courses, as well as the roles and responsibilities of the teaching assistants, are briefly described in the following chapters.

3. Introduction to Computer Science, First Year

The course Introduction to Computer Science (DA100D, 2020)[4] is the first course that first-year computer science students encounter in their academic journey. The course comprises 7.5 higher education credits (HEC) and provides a broad introduction to computer science, laying the foundation for both theoretical and practical understanding as well as academic skills. It also offers students an early perspective on applied computer science and the ethical responsibilities of working as a programmer. Throughout the course, students engage in both individual and group work. They are required to produce a written group report based on their collaborative work, as well as an individual academic report. Students practice academic writing, engage in peer review of each other's reports, and develop their presentation skills during structured seminar sessions.
As this course represents students' first step into academic and scientific study, we place emphasis on fostering early engagement and helping them build a solid academic foundation from the very beginning. Based on our experience with first-year students and continuous course development over

the past five years, we have found that the most effective solution has been to integrate TAs in a mentoring role. Initially introduced during the pandemic, TAs provided online support exclusively through the Discord. However, over time, their role has evolved significantly. Today, TAs play a much more central and visible role in the course. They are available not only in the evenings via online platforms but also on campus during the day. TAs are actively involved in exercises and laboratory sessions, where they support students in understanding complex topics, particularly during the early stages of their computer science education.

In recent years, the number of enrolled students has increased significantly—from around 30 to 130. Managing such a large course requires substantial experience and careful coordination. Currently, one main course coordinator is responsible for both the administration of the course and the majority of the lectures. An additional instructor assists specifically with the assessment of written reports. The course is given for two groups: one international and one national. This adds another layer of complexity to course management, as the coordinator must address diverse student needs and learning contexts.

Given the scale of the course, the role of the Teaching Assistants (TAs) is especially valuable. TAs provide essential support to the course coordinator, helping to maintain the quality of the learning experience despite the increased student numbers. However, TAs do not assess or grade students—their role is solely to support, guide, and encourage learning. Their presence helps ease the transition into higher education and contributes to a more supportive and accessible learning environment for new students.

4. Algorithms and Data Structures, second year

The course Algorithms and Data Structures, (DA256E, 2022) [5] covers theoretical parts of computer science and with mathematical approaches to such theories. The course normally has between 60 and 80 participating students. Furthermore, the course has been provided during the third semester of the bachelor study programs, and besides mathematical-theoretical elements, the course also includes elements to approach the course contents from practical perspectives. Thus, the course includes two main elements as further outlined below:

Theories, that are generally provided through traditional lectures, and which are examined through traditional written exams. The course leader has more or less complete responsibility for implementing this.

Practices, that are provided through seminars. Here, students are obliged to prepare for the seminars in advance where parts of the theories shall be studied, and exercised, typically through programming exercises.

At the seminars, students discuss their solutions in smaller groups under the supervision of teachers. Students are approved on the basis of the completed preparatory tasks and discussions in which they are able to answer for their solutions. The seminars are normally very time-consuming to conduct, and for the sake of both the course leader and the students, the course leader has chosen to enlist the help of TAs.

Typically, a seminar is conducted by the course leader together with two TAs. These TAs go around the student groups and participate in the discussions. When it is seen that students have completed their part of the seminar, they are marked as passed. A TA therefore has the opportunity here to participate in the examination of the practical course element, but only in the form of a passing mark, or failing with the need for later completion.

5. Results and Analysis

5.1. Introduction Course

In 2021, there were two TAs assigned to the course, and since it was still during the pandemic with online restrictions in place, the TA's main role was to support students via the Discord. The interaction on Discord developed organically, with students reaching out primarily when they needed help or clarification. The TAs maintained a clear structure in communication by reserving a "questions & answers" channel specifically for course-related inquiries, while directing off-topic discussions to a separate channel. Students occasionally helped each other by responding in the chat, though this was not frequent.

Since the course is the very first for the first-year students, TAs received a large number of questions outside the course content, particularly related to organizational or administrative matters, many of which were already answered on Canvas or required the course instructor's input. Examples included questions about deadlines, lecture attendance, Zoom links, and formatting issues. This underscored

the importance of clear and ongoing communication between TAs and course instructor, ensuring that TAs are well-informed and able to guide students effectively. It also emphasized the need to provide instructions for TAs, focusing on communication skills to better equip them for their mentoring role. This experience further highlighted the need for broader student support and became a key input in discussions that led to the creation of a new role: “general TAs”, who would serve as mentors specifically for first-year students. This initiative was officially implemented in 2022.

On average, one TA spent about three hours per week on Discord, with most student activity occurring after 4 PM. The TAs found the amount of time they received (50 h each), was sufficient and saw no need for increased presence. There was no formal time division between the two TAs; they simply responded as needed, covering for each other when necessary.

Although one TA could manage the role, having two proved more efficient and manageable, balancing the TA´s workload and providing better support to students. In 2022, when only one TA was assigned, the workload became overwhelming despite being handled effectively. As a result, it is now recommended to always assign two TAs per course to ensure balanced support.

In 2023, the number of enrolled students in the course reached 135, and three TAs were employed. All described the role as highly rewarding, highlighting how it helped reinforce their own knowledge, improved communication skills, and deepened their connection with peers. One of the most fulfilling aspects was the appreciation expressed by students, as one TA reflected, “The gratitude I saw in the eyes of the students I helped”.

Challenges varied, with time management and simplifying complex topics being the most common difficulties. Balancing their own coursework with TA duties was also mentioned as a key challenge. In terms of Discord engagement, approximately 120 students joined the platform, with about 40 actively participating. The TAs noted that interaction developed both naturally and through intentional engagement, including answering questions, sharing resources, and occasional voice calls. On average, they spent 1–2 hours per day on Discord, including weekends, and managed their responsibilities flexibly—responding based on availability rather than having fixed schedules.

Most student inquiries stayed within course content, though occasional questions related to university logistics were also received. All three TAs agreed that the time allocated to Discord support was sufficient, and no additional time was deemed necessary for future iterations of the course.

Overall, the feedback confirms that the TA model not only enhances student learning but also provides valuable personal and academic development for the TAs themselves. Discord has proven to be an effective communication platform, enabling timely support, flexible interaction, and a stronger sense of community.

Student satisfaction with the course remains consistently high, with evaluations from 2020 to 2024 averaging around 4.5 out of 5. TAs play a crucial role in this success, offering support both on campus and online via Discord. They are regarded as knowledgeable, approachable, and responsive—qualities that contribute to effective communication and a supportive learning environment, particularly for first-year students. Student feedback regularly emphasizes the positive impact TAs have on the overall course experience:

“Interesting subjects, knowledgeable teacher and teacher assistants. [Teacher] and [TA] were very enthusiastic of the topics and us learning them.”

“... Also ..the TA [...] ... is always ready to help a student, [...] is very cooperative and has a good knowledge on the subject”

“[...], the TA, was very supportive and helpful throughout the course.”

“In my opinion, practical tasks were very useful, I liked that we could go through the homework tasks together in class and that if something was not understandable, then [Teacher] or [TA] explained in a different way than in the book, for example.”

It is also worth mentioning how senior students typically become TAs. Usually, the course instructor identifies potential TAs during the course itself by observing student interactions and performance, as well as gathering feedback from current TAs and other students. Often, students express their interest in the TA role early on after positive interactions with existing TAs. When the instructor recognizes that these students are engaged, demonstrate strong communication skills, and possess a solid understanding of course content, they are usually invited to become TAs in the following year. One TA described this motivation clearly:

“I wanted to be a TA since day one. Having the chance to help other students is mostly what I am good at. This experience enriched my study years.”

5.2. Algorithm Course

The following addresses the outcomes of the Algorithm course during the fall terms of 2023 and 2024. It is worth noting that, since the course is intended for second-year students and the subject matter is more advanced than during the first year studies, the expectations placed on the TAs are correspondingly higher—requiring greater subject expertise and a more proactive level of support. On these occasions, the course seems to have worked very well, and the students' course evaluations clearly show that TAs have contributed well to this. Examples of student reviews regarding TAs´ involvement include:
The discussions on the seminars were very productive, the communications with the teacher and TAs were good.
More TA sessions for the Seminars
The course was great with the help of the TAs and all. It was a great course.
...Assistant teachers also very helpful.
The reflections from TAs highlight a range of academic, professional, and personal benefits gained through their involvement in course support roles. Their experiences reveal several key themes:

Deepened Subject Understanding:
Both participants emphasized that teaching core concepts, such as algorithm complexity, relational schema design, and iterators, significantly enhanced their own comprehension. Explaining difficult material to others required them to restructure knowledge in accessible ways, reinforcing their grasp of the subject.

Professional Skill Development:
The TA role contributed to growth in essential soft skills, including communication, leadership, and collaboration. Both reflections mentioned improved confidence in public speaking and an increased ability to support students with varying learning styles and needs.

Shift in Perspective and Motivation:
Acting in a teaching capacity offered new insights into the effort and responsibility involved in course delivery. Stepping into the instructor role gave both TAs a greater appreciation for educators and sparked an interest in future roles involving teaching or mentoring.

Recommendation for Peer Participation:
The experience was not only seen as rewarding but also as recommendable. One participant explicitly encouraged other students to pursue TA opportunities to gain both technical and interpersonal skills.
“The TA experience has strengthened my confidence in technical problem-solving, mentoring, and teamwork all of which are essential in the professional world. It has also inspired me to stay open to future roles in teaching or knowledge sharing, whether in academia or industry.
I will recommend TA positions for other students as well for them to gain experience and increase their knowledge and soft skills as I did□”
“As the semester progressed, I found that the act of teaching itself became increasingly enjoyable. Engaging with students and assisting them in grasping complex concepts fostered a sense of responsibility and fulfillment. While I have always admired the professors who taught me, stepping into their role gave me a newfound appreciation for the profession. Indeed, standing in their shoes was a markedly different—and unexpectedly gratifying—experience. In summary, my time as a Teaching Assistant allowed me to grow both academically and personally, and it further solidified my interest in teaching as a potential future path.”
Overall, the reflections illustrate that TA positions provide meaningful experiential learning that benefits both the students receiving support and the TAs themselves—enhancing academic mastery, soft skill development, and professional identity.

6. Summary and Conclusion

This study demonstrates that involving senior undergraduate students as Teaching Assistants in Computer Science education offers clear benefits for students, instructors, and the TAs themselves—particularly in resource-constrained environments like smaller universities. Drawing on both course evaluations and reflections, the findings reveal that TAs contribute significantly to improved communication, peer-to-peer support, and student engagement—especially in large, foundational courses where individual teacher support may be limited.

In the Introduction to Computer Science course, TAs served primarily as mentors and facilitators, with a strong presence both on campus and via Discord. Their role proved critical in assisting first-year students, many of whom sought guidance not only on course content but also on broader academic and organizational matters. This experience led to the formalization of a new mentoring role: the "general TA".

In the more advanced Algorithms and Data Structures course, TAs played a pedagogical support role in seminar-based learning, reinforcing students' understanding of complex theoretical material. Student feedback consistently acknowledged the value of TAs involvement, and TAs themselves reported professional and personal development, including improved communication, teaching skills, and subject mastery.

While it is possible for one TA to handle the workload, the results clearly indicate that having two TAs per course leads to more sustainable support, better communication with students, and a more balanced distribution of responsibilities.

Based on the experiences from both courses, the following recommendations are proposed:

Assign two TAs per course, especially for large or first-year courses, to ensure sustainable support and workload distribution.

Introduce "general TAs" or Mentors early in the program to support students with administrative questions, academic orientation, and course navigation.

Provide instruction for TAs, focusing on communication to equip them for their mentoring role.

Use communication platforms like Discord to increase accessibility and flexibility, especially for evening support and informal peer learning.

By following these recommendations, other institutions can successfully implement and sustain TA programs that enhance both teaching quality and student learning outcomes.

On the Future of TA´s role in the World of AI

As artificial intelligence (AI) continues to reshape education and professional workflows, the role of TAs is also evolving. With the increasing availability of AI-powered tutoring systems, automated feedback tools, and intelligent grading platforms, questions arise regarding how human TAs will remain relevant and how their roles may need to adapt.

TAs play an important role in reducing the workload of teachers, both in the classroom and in communication with students. This is particularly critical with a large number of students when teachers´ resources are limited. However, communication via Discord has declined in favor of students turning to AI tools. Still, TAs continue to play a crucial role in handling questions where AI tools do not always provide accurate or helpful answers. We see the role of TAs as essential in identifying and addressing this type of issue.

Here, rather than viewing AI as a replacement for TAs, it is more productive to see it as a complementary tool that can enhance their effectiveness. For example, AI can be used to handle repetitive tasks such as answering frequently asked questions, providing instant feedback on assignments, or tracking student progress—freeing TAs to focus on more meaningful, human-centered interactions such as mentoring, peer support, and facilitating deeper learning discussions.

REFERENCES

[1] Reges, S., McGrory, J., Smith, J.. "The Effective Use of Undergraduates to Staff Large Introductory CS Courses", Nineteenth SIGCSE Technical Symposium on Computer Science Education (SIGCSE '88), ACM, New York, NY, USA, 22–25. DOI: http://dx.doi.org/10.1145/52964.52971, 1988

[2] Mirza, D., Conrad, P. T., Lloyd, C., Matni, Z., Gatin, A., "Undergraduate Teaching Assistants in Computer Science: A Systematic Literature Review", ICER '19, Toronto, ON, Canada, https://dl.acm.org/doi/pdf/10.1145/3291279.3339422, 2019

[3] Klonowska, K., Chen, E. Z., Kjellstrand, I., Källström, L., Siljeklint, P., "Student engagement in hybrid taught large introduction courses during COVID-19", The Future of Education, FOE 2021 - 11th Edition, 2021.

[4] DA100D, Course syllabus – Introduction to Computer Science, 7.5 HEC, English, HKR.se, Course syllabus - Introduction to Computer Science - da100d , English | HKR.se, 2020.

[5] DA256E, Course syllabus – Algorithms and Data Structures, 7.5 HEC, English, HKR.se, Course syllabus - Algorithms and Data Structures - da256e , English | HKR.se, 2022.