Experiences of Assessment in Introductory Programming From the Perspective of Non-Computer Science Majors

Emma Riese
Theoretical Computer Science
KTH Royal Institute of Technology
Stockholm, Sweden
https://orcid.org/0000-0002-4525-3568

Stefan Stenbom
Digital Learning
KTH Royal Institute of Technology
Stockholm, Sweden
https://orcid.org/0000-0002-8200-6021

***Abstract*— This Research, Full Paper presents a study of how engineering students, non-computer science (CS) majors, experienced the assessments during their first programming course (CS1). Naturally, we as instructors and course designers would like to facilitate learning among our students and make fair assessments. During the CS1 courses given for non-CS majors at our university, the students would be assessed on their programming skills and concept knowledge on multiple occasions. Typically, students need to complete 4-6 lab assignments, pass an exam, and complete an individual project. The instructor in charge of each course, the course coordinator, design all assignments as well as the grading criteria and is typically also responsible for the assessment of students' skills and knowledge. However, since these courses have a large number of students (~200) the assessments are mainly done by teaching assistants (TAs). The purpose of this paper is to explore how students experience the assessment situations in CS1 courses.**

Eleven semi-structured interviews were performed with engineering students who enrolled in one of four CS1 courses. The interviews were transcribed and analyzed using thematic analysis. The results indicate that the students experienced the individual project to be authentic, a fair and reasonable way to assess their programming skills. By contrast, the exam was experienced as inauthentic and focusing on less important skills such as knowing syntax by heart. The students put lots of trust in their TAs, but experienced that the grading and amount of feedback they received, differed depending on the TA. The hierarchy, in which the course coordinator instructed the TAs on how to conduct the assessment, was not clearly visible and some students even viewed their TAs to be more qualified to grade their assignments than the course coordinators. It was, however, clear to the students that the course coordinator was also the course designer and the one who constructed the exam.

Keywords— Assessment, Students' experience, CS1

I. Introduction

Enrolling in your very first computer science course (CS1) and learning how to program can be quite challenging [1, 2, 3] and failure rates in CS1 courses have been reported as high [4]. The transition from high school to higher education has also been shown to be experienced as challenging and students can, for instance, worry about failing courses and not getting enough help from their professors [5]. Naturally, we as instructors and course designers like to facilitate learning among our students and direct the students' learning towards the learning objectives of the courses we teach. In order to do that, the intended learning outcomes, learning activities, and assessments should be aligned, referred to as constructive alignment [6]. The assessments we, as instructors, design can both serve to judge students' performances and to guide and motivate our students towards achieving the intended learning outcomes [7]. Feedback is also an important factor for student achievement [8] but has to be given at the right time [9]. In courses that are given to a larger number of students, teaching assistants (TAs) are commonly used to reduce the workload of the faculty and allow for smaller groups and individual tutoring [10]. The TAs are themselves, students, in our case typically undergraduate or Master's students, taken on the role in parallel to their own studies.

While prior research about assessment in introductory programming has been focused more on the students' grades and validity and fairness of the assessment [11], this study focuses on the much less studied student perspective. The aim of this study is to identify and describe how the assessments throughout CS1 courses for non-CS majors are experienced from the perspective of the students.

The research questions addressed in this paper are:

1. How do students experience the assessment of different types of assignments (lab assignments, exams, and projects) in their on-campus CS1 courses?
2. What are the students' perceptions of the TAs and lecturer/course coordinator in relation to the assessments?

The results of this study could be of interest to faculty and TAs who are teaching, or planning to teach, in CS1 courses.

II. Related Research

Teaching someone how to program has been reported as hard for decades [1, 2, 3]. The assessments throughout a course can guide and direct the students to achieve the intended learning outcomes, and how well the students fulfill the learning objectives is also what should be reflected in the

students' grades [7]. Assessment is commonly divided into two parts, formative assessment and summative assessment [7, 12]. Formative assessment serves the purpose of providing feedback and guidance, assessments conducted to assist and support the learning process [7,12]. Summative assessment on the other hand is an assessment conducted with a focus on measuring and judging how well a student fulfills a certain learning objective, master a skill or shows certain knowledge [7, 12]. Even though formative assessment and feedback are important, organizing extra voluntary learning opportunities has not always proven to be successful, since students might not see what is in it for them and pass up on the opportunity [13].

To facilitate learning towards the intended learning outcomes, the learning objectives, learning activities, and assessments should also align, called constructive alignment [6]. When grading in higher education, a common approach for assessment is the so-called *analytic approach* [14]. An analytic approach to grading is conducted by using grading criteria and grading rubrics as a basis for the assessments [14]. A literature review on grading rubrics in higher education showed that instructors believe using rubrics keeps the grading more objective, however, it is also concluded that some instructors remain unwilling to use them [15]. For successful use of grading rubrics, the language used in the rubrics needs to be understandable to both the students and the graders, since an unclear language can make it possible for different interpretations [15].

In addition to conducting assessments, the teachers or TAs can, and should, also be a facilitator of learning during the learning activities. Vygotsky defined the Zone of Proximal Development (ZPD) as the difference between what a learner can do by themselves, and what they can do with the help of more capable or skilled persons [16]. This implies that for a learner to be able to solve or accomplish a certain task, it should be within their ZPD. If the learner can solve it completely by themselves, it could be a too easy task.

Introductory programming is a well-researched area and there are lots of previous studies that have been conducted regarding assessment in courses for novice programmers [11]. The authors of a quite recently conducted literature review concluded that much of the research on assessment approaches have focused on student performance (grades) and the validity and fairness of the assessments [11]. Quite many studies have also been conducted on tools for providing students with automated feedback [17].

An assessment method developed to test students' programming skills, regardless of programming language, that was tested on multiple institutions, resulted in poor results among the students [18], indicating that students who enrolled in CS1 did not master the expected skills and knowledge. It has been proposed that multiple-choice questions are a well-suited way to assess basic knowledge and comprehension in CS1 [19]. How students experience assessment has also been studied before in [20] where a phenomenographic research approach was used to identify experiences of assessment for students within an online introductory CS course. Five categories were found: the grading is important to the teacher, the grading is important to the student, assessment as guidance, assessment as an opportunity to learn and assessment as a way to communicate. [20]. The author concluded that three categories, assessment as guidance, assessment as an opportunity to learn and assessment as a way to communicate, are the most desirable [20], but did not do any distinction between how the different assessment types were experienced.

Another important perspective is that assessment in CS should be authentic, similar to the practitioners, but at the same time relevant for those who study it [3]. One successful example of how CS1 courses were experienced as relevant to the students, was through adapting the content to the students enrolled in the course, by introducing "media computation" [21]. A previous study on how first-year computing students experience authenticity, showed that the students had a reasonable perception of what authentic assessment was, that the assessment should be linked to real-world situations [22]. The student's view of authenticity was also later compared to the faculty's definitions and found to align quite well [23]. A literature review of teaching intervention and their effect on student's pass rates in introductory programming, showed that teaching interventions, such as incorporating collaborative or peer-support, and making the course content relatable to the students, on average had a positive effect on the pass rates [24].

In addition to the instructors (course coordinators and lectures), TAs have been used in CS1 courses for many years and by many institutions [10, 25]. To employ TAs has been a way to reduce the workload of faculty and by the same time being able to scale up courses [26, 27]. The TAs work tasks and responsibility differs between institutions and courses but include grading assignments, tutoring, conducting tutorials and grading exams [10]. TAs have been shown to play an important role in student success [28], but there is also evidence that the TAs could have received little training and are poorly prepared for their responsibilities [29, 30, 31].

III. Research Setting

At the university where this study took place, multiple 5-years programs towards a Master of Science in Engineering degree are offered. During the students' freshman year, the students enroll in a mandatory CS1 course, where the programming language Python is used. Most of the educational programs have their own edition of the CS1 course, allowing the content to be tailored to fit within their program and choice of major. For this study, we interviewed students who majored in the following: engineering physics, industrial economics, mechanical engineering, and students enrolled in a special engineering program who declare major after their first year. The intended learning objectives are similar between all of the courses; the students are expected to learn basic programming and be able to construct their own programs. This includes how to use data structures (such as lists), construct and use classes and objects, functions and methods, follow praxis when writing code and be able to debug code. For the engineering physics students, the learning objectives also included recursive functions, handling versions in Git and being able to describe how data is stored by Python. The course grades were given on the scale A-F, with A being the highest grade, E the lowest passing grade, and F a failing grade. All of these CS1 courses are also similar when it comes to the set-up of the assessments. All four courses begin by having a number of smaller lab assignments (4-6), followed by a written exam, and finally an individual project. Each of these assessment types is described in detail in the following three subsections. An overview of the courses' structures and the differences between them is shown in Table 1.

TABLE I. OVERVIEW OF COURSES

	Courses			
	Engineering physics	***Industrial economics***	***Mechanical engineering***	***Not declared major***
Study period	10 weeks	20 weeks	20 weeks	20 weeks
Course credits (ECTS)	5	7.5	6	6
Number of lab assignments	4	6	6	6
Grading scale used for lab assignments	Pass/fail, but with extra tasks and the requirement to meet the deadlines for higher grades.	Pass/fail	Pass/fail	Pass/fail
Presentation of lab assignments	Orally but also submitted through Git	Written and feedback given in written form	Orally	Orally
Bonus points towards exams	No	Yes, collected by quizzes	Yes, collected by meeting the deadlines for the lab assign-ments	Yes, collected by meeting the deadlines for the lab assign-ments
Exam	Pen and paper, 35 questions, divided into 7 sections	Computer lab, 10 questions with fill in code gaps	Computer lab, 20 multiple choice questions	Comput-er lab, 20 multiple choice questions
Project (orally presented)	Graded A-F, including specific-ation, peer-review	Graded A-F, including specifica-tion, and peer-review	Graded A-F, including specifica-tion, and peer-review	Graded A-F, including specifica-tion, and peer-review
Additional assess-ments	No	Yes, a short MATLAB part, excluded from the study.	No	No

A. Lab Assignments

Each course has lab sessions scheduled in the computer labs, where the students would work on their given assignments. During the lab sessions (typically 2 hours long) TAs were present and could assist the students in their work. The courses used a digital queuing system where the students can request help, and the TAs can keep track of who is next in line. In one course, the one for majors in industrial economics, the students were also divided into subgroups where a TA was responsible for the same group of students throughout the course. In the three other courses, there were no set groups and the TAs helped the student who happened to be first in line when they were available. The lab sessions were in general voluntary for the students to attend. However, depending on the course, the students needed to either present their assignment's solutions orally to a TA during one of the lab sessions or by submitting code in the learning management system (LMS) or platform used. The students from all courses could choose to either work in pairs or individually on the lab assignments. The students needed to pass all lab assignments in order to pass the course. In all courses except the one for students majoring in engineering physics, all assignments were graded on the scale pass/fail. In the course for engineering physics, the set-up was slightly different and the students who aimed to get a higher grade than a D, needed to complete some extra assignments and also meet the deadlines for the assignments. In two of the other courses, the one for the mechanical engineering students and the one for students who have not yet declared their major, the students could be awarded bonus points that could be used on the exams (to lower the requirements for passing the exam) by presenting their assignments on or before a set deadline. In the course for industrial economics students, the students could be rewarded bonus points by weekly quizzes given during the tutorials.

For the assessment of the lab assignments, the TAs would typically use the same instructions and requirement lists given to the students (a simpler version of grading criteria).

B. Exam

In all four courses, part of the examination was done through a written exam. In each of the courses, the exam was graded on a scale pass/fail and the students needed to pass the exam in order to pass the course. The exam was very similar in the two of the studied courses, the one for mechanical engineering students and students who had not yet declared majors. In these two courses, the exams took place in a computer lab, where the students were given a set of 20 multiple-choice questions and needed to answer 16 correctly inorder to pass the exam. The students were also allowed to bring a textbook to the test if they wished. In the course for industrial economics students, the test also took place in the computer lab, but the students were instead given a set of 10 unfinished Python-program snippets and asked to fill in the code that was missing. The students also had the possibility to test-run code while answering the questions. In the course for engineering physics majors, the exam was a written exam using pen and paper. It was constructed into seven parts (mapped to each learning objective of the course), each part consisting of five questions. To pass the exam the students need to get at least half of the points for each of the seven parts. These students were also allowed to bring their textbooks to the exam.

C. Individual Project

The final part of the courses consisted of a larger project, an individual programming assignment. The individual project was assessed using the scale A-F, ultimately deciding the students' course grade (in the course for engineering physics students, they would also have to finish the extra task on the lab assignments and meet the deadlines to receive higher grades). The students had a large set of project tasks to choose from, each slightly different from each other. There was, however, a restriction on how many students could pick the same project. Each project consisted of the basic assignment for grades E, D (for D, the structure of the program has to be better than what is acceptable for E) and additional extra tasks that were more complex and harder to solve for grades C, B, and A. For example, to receive the grade B the students would have to complete the basic assignment and the additional tasks for C and B. The project consisted of three sub assignments, a specification in which the students had to

present an approach to solving the problem, a peer-review step, and a final presentation. In the peer-review step, the students would work in pairs and review each other's code following a protocol with code specifications, such as requirements of code structure, readability of the code and usability of the program. The reviewer, as well as the developer of the program, would then be present at the presentation. During the presentation, the reviewer would present their review and then the TA conducting the assessment would ask questions to the students and grade the project. The TAs would also use a similar grading rubric (protocol) as the one used in the peer-reviewed assignment, which includes requirements of code structure (has to have a class), readability of the code (variable names and comments needs to be appropriate) and usability of the program (the user should understand what to do and what is presented on the screen). To receive a passing grade (E-A) the students' code needs to fulfill all those requirements. The TAs would also have access to the instructions for each project, which states the functionality requirements for each of the grades E-A.

IV. Method

Eleven semi-structured interviews were performed with engineering students who had participated in one of the four CS1 courses during the fall semester of 2019. This qualitative method was chosen because we wanted to get a detailed picture of the students' experience without having to base any questions on assumptions. In this study, we are not necessary searching for a generalizable truth, but rather a detailed picture from a smaller number of respondents.

Each course was slightly different depending on the students' majors, as described in the previous section. The courses were all given during the students' first semester as a mandatory part of their engineering program. The students were recruited to the study by advertising on the course web shortly after each of the courses had ended and students had received their course grades. The course coordinator of each of the courses assisted in publishing the information but was not part of the study in any other way. The students were given a short summary of the purpose of the study, who were conducting it and an approximation of the time commitment required to participate.

All interviews were conducted by the first author of this paper who had no prior relation to any of the students, and was neither a course coordinator nor a TA in any of the courses the students had enrolled in. Six of the students were engineering physics majors, three students majored in industrial economics, one in mechanical engineering and one had not yet declared major. It was completely voluntary for the students to take part in the study. All participants were given detailed information about the scope and aim of the study. They were also reminded that participation was completely voluntary and that they at any time could decline to answer a question or determine their participation without being questioned for doing so. The participants were also informed about how the collected data was going to be handled and by who (only by the participating researchers in the raw form). All participants gave their consent to be part of the study and agreed to have the audio of the interview recorded. As a small token of appreciation, all students who participated in the study received a cinema ticket gift card.

The set questions for the interview were:

- *Would you like to start by telling me about the course, how has it been organized?*
- *Which were the assessments?*
- *How has it been to participate in the course?*
- *How does one present/ submit a lab assignment?*
- *Have you worked in pairs or alone with lab assignments?*
- *How did you experience the assessment of the lab assignments? (summative assessment)*
- *Did you get any feedback on something you could use later in the course? (formative assessment)*
- *How did you feel about getting help from a TA?*
- *How was the exam structured?*
- *How did you experience writing the exam?*
- *How did you experience the assessment of the exam?*
- *If you think about the project, would you like to describe the different steps in the project?*
- *How did you think it was to first plan your project and write a specification?*
- *What kind of feedback did you get during the project work?*
- *How did you think the peer-review worked?*
- *How did you experience the project presentation and assessment?*

In addition to the above questions the interviewer asked each respondent a number of follow-up question depending on their answers. This was done to allow the respondents to further explain and elaborate on their answers, which led to a more detailed data set. The interviews were transcribed and analyzed using thematic analysis [32]. The analysis was conducted by following the six steps in [32], however, as a first step, the data was divided into four parts, one part for each assessment type (lab assignments, exam, individual project) and the fourth one for data sequences that fitted neither of the first three. The thematic analysis was performed by the first author of this paper and validated by the second author. The interviews were conducted in the native language in the country of the study. The illustrative quotes that are presented in the paper has been translated to English by the authors.

V. Results

The results are structured around the two research questions. First, the students' experiences of the different assessments used in the courses is presented. This is followed by the students' perception of the different teachers' roles present to support and assess students' learning in on-campus CS1 courses.

A. How are different types of assessment experienced?

In this section, the results that relate to the first research question is presented. The research question is: How do students experience the assessment of different types of assignments (lab assignments, exams, and projects) in their on-campus CS1 courses? The section consists of three main subsections, one for each assessment type: lab assignments, exam and individual project. Under each subsection, the identified themes are presented with some example quotes from the interviews.

1) Experiences of the lab assignments: The thematic analysis resulted in two themes regarding the students' experiences of lab assignments: *lab assignments as a necessary evil* and *lab assignments as a learning activity*. Both themes are described in more detail below, and an example quote for each theme is also presented.

a) Lab assignments as a necessary evil: Many of the interviewed students, especially those who had little or no prior programming experience, experienced the timeframe with continuing tight deadlines as stressful. This contributed to the fact the students wanted to just finish the assignment, rather than to take their time to learn the concepts and tools needed. The shortage of time was also described as a contributing factor to an uneven distribution of work between the students working in pairs. The main goal was to pass the assignments, and the fastest way to reach that was by some respondents described to be that one person took command and left the other one behind.

"Yes, well it almost played out in such a way that he [lab partner] did it [wrote the assignment code] and then explained it to me. And it's like .. well you learn from doing. So many of those who were true beginners, working together, they learned a lot more than I know I, and others who also ended up with pretty skilled partners, did." - Industrial economics student

b) Lab assignments as a learning activity: The lab assignments were also percived by the students as a way to practice and learn the concepts that had been brought up in the lectures. The students' could acknowledge that this was the thought behind having the lab assignments, but that it had not always played out like that from their perspective. Some of the students expressed that they were already good at programming and for them, these assignments were easy, while for the novice students they could be experienced as very hard.

"The idea was, I suppose, that you should learn. That one would learn the basics of Python through the labs and then one would also remember that [what you learned] towards the exam." - Industrial economics student

2) Experience of the exams: The thematic analysis resulted in two themes regarding the experiences of the exams: *predictable but not suitable* and *inauthentic assessment of programming skills*. Both themes are described in more detail below, and an example quote for each theme is also presented.

a) Predictable but not suitable : The students were all used to having written exams but to many of the respondents, this was the first exam in CS. For most students, it was also the first exam graded on scale pass/fail. Compared to the lab assignments and the project, the students experienced the exam to be a more unsuitable way to assess their programming skills and the knowledge, that they viewed as relevant. At the same time, the students could appreciate the known format and that they could study on previous exams. The students experienced the exam to not be directly linked to the other assessments and a student who received an A on the project could have failed the exam. In the course for engineering physics, advertisements of previous low passing rates had caused the students to worry and for some, the requirements were experienced as almost impossible. Since these exams differed in the way that each subset of questions was linked to learning objectives, the students could, however, see the reason for the design.

"We were given a practice exam, like the day before the exam or maybe two days before, and then you could check that out and see well I should focus on studying this and that, and then you knew that on the day of the exam. So I think the practice exam was very important, to just understand what types of questions can be asked." - Engineering physics student

b) Inauthentic assessment of programming skills: The exam was experienced as inauthentic and focusing on less important skills such as knowing syntax by heart. The students questioned how the skills being assessed in the exam would even matter in real life since if they were to write code they would do it on a computer, have access to the internet and be able to test run their code.

"Some problems are never encountered when programming, but they are nevertheless examined. But I sort of have an understanding of how the code works when it runs. But there were questions like how many syntax errors should this result in? OK, it wasn't really too much of that, I had heard it was worse [from the older students]. There was a question where you thought it was going to check one thing, a certain objective, but then it was something else, so it was the wrong thing." - Engineering physics student

3) Experience of the individual projects: The thematic analysis resulted in four themes regarding the experiences of the individual project: *a leap from the lab assignments*, *learning by doing*, *fair and authentic assignment and assessment* and *difference in difficulty between projects*. The themes are described in more detail below, and an example quote for each theme is also presented.

a) A leap from the lab assignments: The students who had worked in pairs on the lab assignments experienced it to be quite a difference to suddenly be by yourself, especially for those who had been leaning on their partner for the lab assignments. All students did also experience the projects to be on a more complex level than the lab assignments and for some it was experienced as a big leap.

"Well, I would have been better prepared for the project if I had to do the labs myself." - Industrial economic student

b) Learning by doing: The students stated that in addition to being graded on the project, working with these larger programming tasks had been a great learning opportunity. During the project, the pieces started to fit together and the students could also see a use for what they

learned during the course. The students also expressed a certain sense of pride by having developed a more complex program that worked according to the specification.

"I think I learned the most from the project. Before, on the labs, it was pretty much given how your code should look like. In the project it was more like 'here is the problem. Have a good time'". - Engineering physics student

c) Fair and authentic assignment and assessment: Overall the students described the project to be an authentic situation, a fair and reasonable way to assess their programming skills. The projects were also assessed using a grading rubric, similar to (or the same) the students used in the peer review-step.

"I think this project task, that's the best way to test, what I consider to be programming. It is not to test "what does this do, how does the code behave", that is not relevant, but it is to solve the problem. So the best way to assess programming, in my opinion, is to present a problem, say that your solution needs to fulfill this and that." - Mechanical engineering.

d) Difference in difficulty between projects: The students did compare the instructions of the project assignments and came to the conclusion that they were not all of the same difficulty. The students had been given advice from sophomore students on which projects to choose and students who already had prior experience could also pick one of the easier projects as a tactic move. For the novice students, picking a project of the right difficulty was experienced as hard, and some of them regretted their choice.

"So if I think about how many hours I spent, and who was there with me [in the computer lab] and used to be there. Then it was really, well beginners who kind of just panicked all the time and just 'no, no! I should have switched [project] and this is not good'." - Engineering physics student

B. Students Perceptions of TAs and Lecturer/Course Coordinator

In this section, the results that relate to the second research question is presented. The research question asked is: What are the students' perceptions of the TAs and lecturer/course coordinator in relation to the assessments? The section consists of three main subsections: one for the course coordinator, one for TAs, and one for informal tutors and course material**.** Under each subsection, the identified themes are presented with some example quotes from the interviews.

1) Role of the course coordinator: The thematic analysis resulted in three sub themes regarding the course coordinator's role, each presented below**.**

a) The designer of the course: The students who expressed any kind of dissatisfaction with how the courses and the different assessments had been handled, mainly blamed the course coordinator. Students described that the course coordinator set the ground rules for how everything should be conducted, stated how many hours they were expected to put into the project and the one who set the deadlines, and also adjusted them if needed.

"It might be very difficult for people to do it themselves. Because, well .. many maybe have not even played computer games in their lives and they think it is very new, complicated to think in the abstract way, that I mean programming is, and view it as impossible. Which, I mean you talk to all your classmates and it was really like this ... people who sat and cried in tears because they were so stressed over the labs. And it was not over the difficulty. I blame the course coordinator." - Engineered physics students

b) The constructor of the exam: The interview students were not too happy about how the exam questions were constructed, and it was clear that the students knew the course coordinator constructed the questions. This was especially the case for questions where the student was expected to hand in a written code without the possibility to test run it before submitting the exam.

"So you have to get everything exactly right, but that is not the case when one is programming. Not even the teacher. When he programmed and show us simple, for him simple programs, he made lots of small mistakes all the time. And if he had written the exam, he would have made quite a few mistakes as well. - Student who had not declared major

c) Somewhat absent: Naturally, the course coordinator in a course for up to 200 students can not keep track of each individual student and the students expressed that the course coordinators were absent from the learning activities that were not lectures. Some course coordinators were described to be hard to get a hold on through emails when the information that was given could be experienced to lack details needed.

"I would say that the teacher, well he left us a little. [...] We submitted the [lab assignments] via the LMS. Then it was the assistants who corrected it. Well, that's it, I don't really know what the teacher did." - Industrial economics student

2) Role of the TA: The thematic analysis resulted in four sub themes regarding the TA's role, each presented below**.**

a) Variation in quality depending on the TA: The students put lots of trust in their TAs, but experienced that the grading, amount of feedback and help you received differed depending on the TA. Some TAs were described as very strict, someone who they tried to avoid, and some to be very kind and helpful. Some TAs were also experienced to have insufficient programming skills or to be unable to go back to the mindset of a novice programmer.

"There was a huge difference in the quality of the TAs. Some could just sit down and say it's approved [the lab assignment] and some TAs could ask you to fix things that

were not in the instructions because they just think so themselves." - Engineering physics student

b) Too few TAs: Almost all of the interviewed students stated that the waiting time to get help had been too long during at least some of the lab sessions. They had rushed to go to the computer lab to get a chance to be in the top segment of the line when the sessions started, and some of them had spent more or less the entire lab time in the queue. The students also picked up that the TAs could be very stressed when the queue was long, and then the students might not even try to get help from the TAs because of it.

"But they were very few TAs in relation to how many students there were. So it was a very long queue." - Engineering physics student

c) Helping hand?: The TAs were described as the persons who helped and guided the students with the lab assignments, towards the exam and with their individual project. However, some of the interviewed students were concerned with the lack of pedagogical skills of the TAs, stating that even if they aimed to help their capability of providing guidance were limited. Some students would remember particular TAs as very helpful, and there were also situations in which the students were friends with the TAs and could reach out to them after class hours. Some students were also a bit disappointed about the amount of feedback they got.

"I know it takes a lot of time for the TAs to keep doing that [give feedback] so I understand that they are not doing it. But of course it would have been nice to get some professional feedback." - Student in industrial economics

d) The TA as the person setting the grades: The TAs were described as the persons who ultimately had the power to decide whether or not a lab assignment was good enough to be passed. Furthermore, the TAs did also set the grade on A-F for the final individual project assignment. Even if the course coordinator was viewed as responsible for the course, the hierarchy, in which the course coordinator also should instruct the TAs on how to make the assessment, was not clearly visible to the students. Some students even viewed their TAs to be more qualified to grade their assignments than their course coordinator. The TAs were also described as the ones who actually knew how the students were doing, something they thought the course coordinator did not always keep track of.

"And then when you're done, you present it to some lab assistant. Then you should be able to account for everything you have done in the code. They tell you whether you passed or not. Well, that's it really." - Engineering physics student

3) Informal tutors and course material: Even though the research questions were to answer how the students perceived the TA's and course coordinator's roles, analyzing the results it became apparent that two other actors play important parts during the course work: other students that serve as tutors and the possibility of finding solutions online. These two identified subthemes are presented below.

a) Informal tutors - ask your peers and older students: The innterviewed students said to rely heavily on each other's help throughout the courses. In two of the courses, the students described how older students who were not TAs in the courses, voluntarily offered to tutor students outside class. The students who stated to have prior experience in programming also stepped up and helped their peers. Some of the novice students also stated that they rather asked friends for advice, than asking their TAs.

"Sometimes I asked the assistants, sometimes I asked friends, it goes a little faster [to ask friends] as well and often .. they tend to be a little better at explaining too. [...] they have like the "beginner" mindset. They have the same point of view as I do." - Student who had not declared a major

"I went there [to the lab session]mostly to help others. It was because I was done myself, so I became like an extra TA for my group. " - Industrial economics student

b) Search the internet for answers: The students' stated that they could get advice from the TAs to search for examples and explanations on the internet. Some students' also said that they preferred to search the internet for answers rather than asking the TAs for help since it was typically much faster than waiting in line. Students who had prior knowledge in programming also acknowledge that to be able to search for and find code snippets which solved similar kinds of problems is a skill you needed to master to be a good programmer.

"Yes you can ask for help and such. I did it sometimes but the answer was almost always that I should google it myself. So I used to do that." - Engineering physics student

VI. Discussion

The purpose of this paper was to identify and describe how the assessments throughout CS1 courses for non-CS majors are experienced from the perspective of the students. The empirical data consisted of eleven semi-structured interviews with students from four different CS1 courses (depending on the students' majoring subject) at one university. The research questions addressed were: (RQ1) How do students experience the assessment of different types of assignments (lab assignments, exams, and projects) in their on-campus CS1 courses?; and (RQ2) What are the students' perceptions of the TAs and lecturer/course coordinator in relation to the assessments?

The results indicate that the students view the lab assignments and individual projects as relevant and valuable both from the perspective of learning how to program and as a way of assessing their programming competence. The students did, however, point out that it is crucial that the tasks are designed in such a way that they support their learning process and are of the right difficulty, within their ZPD [16].

If not, the tasks can be perceived as a necessary evil rather than a learning opportunity. Furthermore, it is important that the assessment of the lab assignments and the project is also viewed as formative, in order to facilitate the student's further learning and that the summative assessment is calibrated to be reliable [7, 12].

The exam is, on the other hand, questioned by many of the students participating in this study due to the fact that it is considered to be an inauthentic measurement of their programming abilities. As pointed out in [19], to use multiple choice questions could be a way to measure comprehension and knowledge, but the assessments should align with learning objectives in the course. Our students seem to view the type of knowledge tested in the exam (that they knew the syntax, could debug and understand code) as not relevant, even though this aligned with some of the learning objectives in the courses, which has be emphasized in previous research [6]. It has been highlighted that CS assessment should preferably be linked to real-world situations in programming and be relevant to the students [3, 21, 22]. The interviewed students put a strong emphasis on the fact that the assessment should be authentic and the exam was not experienced to be such and therefore viewed as an unsustainable way to test their knowledge.

Regarding the second research question about the role of TAs, and course coordinators, it is clear that the students perceive the TAs to be their main teachers, the ones most present in their courses. The interviewed students consider the course coordinator to be the person that has designed and organized the courses before the courses start but are somewhat absent during the course itself, while the TAs, on the other hand, are the ones facilitating students learning and grading assignments. The students describe that there is a difference in the quality of the TAs, which would indicate that the TAs are not properly trained or prepared for their work tasks, in line with previous findings [29, 30, 31]. In some previous studies, it has been stated that TAs are more approachable for the students and that the TAs can easier understand and relate to the novice students since they were themselves novice quite recently [10]. Our results do, however, indicate that this is not necessarily the case and that from the perspective of the students the TAs can also struggle with understanding the mindset of a novice student. The more alarming result concerning the TAs' roles is that the students expressed the TAs to both make subjective decisions when it comes to grading and that some of the TAs were not helpful and could even seem to lack content knowledge. This could be an indicator that we need to review how we recruit TAs, how we train them and what type of instructions they are given. The students also experienced the TAs to be stressed and too few, so that the students cannot get all the help they need. The more surprising result, at least to the authors of this paper, was that informal tutors, older students who are not TAs, played a big role in the students' learning process and that they found quite structured ways to help each other. Given the result that the TAs were too few, it is on the other not surprising that students sought other ways to receive help. It might also be of value to introduce a more rigorous peer support system in the design of the courses as it has been found to have a positive effect on pass rates [24]. How the students use other resources when search for answers, could also be something to look closer at, since it might cause plagiarism.

A. *Trustworthiness*

It was completely voluntary for the students to participate in the interviews. It is possible that the students who choose to participate in the interviews were the students who had the strongest opinions about the courses and saw their participation as an opportunity to make their voices heard. The sample size is small, not necessarily generalizable to the whole student population but that was not the aim of the study. This study should rather be seen as the first step towards gaining a more nuanced picture of how the assessment situations during a CS1 course for non-CS majors are experienced by the students. The majority of the interviewed students were engineering physics students and their course differentiated a bit from the other three courses. It is possible that there were differences between the experiences based on the courses. However, in this study, we have not aimed or tried to account for differences that depended on the course designs. The learning objectives and assessment types used are viewed as similar, and we have therefore only conducted the analysis for the whole group. To be able to investigate the difference between the courses, a larger sample would be required.

B. *Implications and lessons learned*

It is alarming that students experienced the requirements, assessment criteria and help received to differ so much between our TAs. These results indicate that we need to prepare and train our TAs better for their role as a grader and as a tutor, similar to what was found in [29]. We also need to design our courses in such a way that our students do not feel left alone, with an absent course coordinator and few and stressed TAs. Staffing the lab sessions with a higher students-TA ration seems like an appropriate next step. Even though it could be desirable for our students to collaborate and work together the students should not feel that the courses are structured in such a way that it is hard to receive help, which has previously been shown to also be a concern among freshmen students [5]. Learning actives that take place outside our classrooms can, naturally, not be monitored. Students should, however, be properly informed about which types of collaborations that is encouraged (for instance discussing and scaffolding the assignments) and which types would be viewed as cheating (copying each other's' code or having an older student writing the code for them).

VII. CONCLUSIONS

In this paper, students' perspective of assessment in CS1 courses for non-CS majors has been examined. Findings indicate that lab assignments and individual projects are seen as relevant and valuable assessment methods by students while the exams are not seen as authentic. This is explained by the CS1 students, as not seeing the link between exams with syntax focus and real-world situations in programming. Furthermore, the students interviewed for this study clearly sees their TAs as more present in relation to assessments, than the lecturer/course coordinator. The TAs are the ones that our students actually meet face-to-face for assessments of lab assignments and projects while course coordinators create the assessment activities and set the grading rubrics. In further research, it will be necessary to justify whether these findings sustain in large-scale studies.

REFERENCES

[1] B. Du Boulay, "Some Difficulties of Learning to Program," Lawrence Erlbaum Associates, 1988. pp. 283–299.

[2] E. Soloway, "Learning to program = learning to construct mechanisms and explanations," ACM 29, 1986, 9, pp. 850–858. http://dl.acm.org/citation.cfm?doid=6592.6594

[3] M. Guzdial, *Learner-centered design of computing education : research on computing for everyone*, 2016.

[4] J. Bennedsen, and M. E. Caspersen, "Failure rates in introductoryprogramming," SIGCSE Bull. 2007, 39(2), pp. 32–36.

[5] M. Zarb, B. Alshaigy, D. Bouvier, R. Glassey, J. Hughes, and C. Riedesel, "An international investigation into student concerns regarding transition into higher education computing," in *Proceedings Companion of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education* (*ITiCSE 2018 Companion*). Association for Computing Machinery, New York, NY, USA, 2018, 107–129. DOI:https://doi.org/10.1145/3293881.3295780

[6] J. Biggs, "Enhancing teaching through constructive alignment," *Higher education*, 1996, 32(3), pp. 347-364.

[7] M. Taras, "Assessment–summative and formative–some theoretical reflections," *British journal of educational studies*, 2005, 53(4), pp. 466-478.

[8] J. Hattie, "Identifying the salient facets of a model of student learning: a synthesis of meta-analyses," *International Journal of Educational Research*, 1987 , vol. 11, pp. 187-212.

[9] J. Hattie, and H. Timperley, "The Power of Feedback," *Review of Educational Research*, 2007, 77(1), pp. 81-112.

[10] D. Mirza, P. T. Conrad, C. Lloyd, Z. Matni, and A. Gatin, "Undergraduate Teaching Assistants in Computer Science: A Systematic Literature Review," in *Proceedings of the 2019 ACM Conference on International Computing Education Research - ICER '19*, 2019, pp. 31–40.

[11] A. Luxton-Reilly *et. al.* "Introductory programming: a systematic literature review," in G. Rossling, & B. Scharlau (Eds.), *ITiCSE 2018 Companion - Proceedings Companion of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education:* July 2–4, 2018, Larnaca, Cyprus, pp. 55-106,. New York NY USA: Association for Computing Machinery (ACM). https://doi.org/10.1145/3293881.3295779

[12] D. D. Dixson, and F. C. Worrell, "Formative and Summative Assessment in the Classroom" *Theory Into Practice*, 55(2), 2016, pp.153–159.

[13] Å. Cajander *et al*., "Unexpected student behaviour and learning opportunities: Using the theory of planned behaviour to analyse a critical incident," *2017 IEEE Frontiers in Education Conference (FIE)*, Indianapolis, IN, 2017, pp. 1-8.

[14] D. R. Sadler, "Transforming Holistic Assessment and Grading into a Vehicle for Complex Learning," *Assessment, Learning and Judgement in Higher Education*, Dordrecht: Springer Netherlands, 2009, pp. 45-64.

[15] Y. M. Reddy, and H. Andrade, "A Review of Rubric Use in Higher Education," *Assessment & Evaluation in Higher Education*, 2010, vol. 35, no 4, pp. 435–448.

[16] L. S. Vygotsky, "Mind in society: The development of higher psychological processes," Cambridge, MA: Harvard University Press, 1978. (M. Cole, V. John-Steiner, S. Scribner, & E. Souberman, Eds. and Trans.; original works published 1930–1935).

[17] H. Keuning, J. Jeuring, and B. Heeren, "Towards a Systematic Review of Automated Feedback Generation for Programming Exercise," in *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education* (*ITiCSE '16*). Association for Computing Machinery, New York, NY, USA, 2016, pp. 41–46. DOI:https://doi.org/10.1145/2899415.2899422

[18] M. McCracken, V. Almstrum, D. Diaz, M. Guzdial, D. Hagan, Y. B-D. Kolikant, C. Laxer, L. Thomas, I. Utting, and T. Wilusz, "A multi-national, multi-institutional study of assessment of programming skills of first-year CS students," *SIGCSE Bull.* December 2001, 33(4), pp. 125–180. DOI:https://doi.org/10.1145/572139.572181

[19] R. Lister, "Objectives and objective assessment in CS1," in *SIGCSE Bull.* February 2001, 33 (1), pp. 292–296. DOI:https://doi.org/10.1145/366413.364605

[20] E. Riese, "Students' Experience and Use of Assessment in an Online Introductory Programming Course," *5th International Conference on Learning and Teaching in Computing and Engineering. (LaTiCE 2017)*, 2017, p. 30-34.

[21] L. Rich, H. Perry, and M. Guzdial, "A CS1 course designed to address interests of women," in *Proceedings of the ACM SIGCSE Conference*, 2004, pp. 190–194, 2004.DOI:10.1145/971300.971370.

[22] R. McDermott, M. Zarb, M. Daniels, and V. Isomöttönen, "First Year Computing Students' Perceptions of Authenticity in Assessment," in *Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '17)*. Association for Computing Machinery, New York, NY, USA, 2017, pp. 10–15. DOI:https://doi.org/10.1145/3059009.3059062

[23] R. McDermott *et al*., "The authenticity of 'authentic' assessment some faculty perceptions," in *2017 IEEE Frontiers in Education Conference (FIE)*, 2017, vol. 2017, pp. 1–9.

[24] A.Vihavainen, J Airaksinen, and C. Watson, "A systematic review of approaches for teaching introductory programming and their influence on success," In *Proceedings of the tenth annual conference on International computing education research* (*ICER '14*). Association for Computing Machinery, New York, NY, USA, 2014, pp. 19–26. DOI:https://doi.org/10.1145/2632320.2632349

[25] E. Roberts, J. Lilly, and B. Rollins, "Using undergraduates as teaching assistants in introductory programming courses: an update on the Stanford experience," *SIGCSE Bull.* , 1995, 27(1), pp. 48–52.

[26] J. Forbes, D. Malan, H. Pon-Barry, S. Reges, and M.Sahami, "Scaling introductory courses using undergraduate teaching assistants," in *Proceedings of the 2017 ACM SIGCSE Technical Symposium on computer science education*, 2017, pp. 657–658.

[27] D. Leyzberg, and J. Lumbroso, "Nailing the TA interview: using a rubric to hire teaching assistants," in *Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education,* 2017.

[28] S. E Dechenne, N. Koziol, M. Needham, and L. Enochs, "Modeling Sources of Teaching Self-Efficacy for Science, Technology, Engineering, and Mathematics Graduate Teaching Assistants," *Cbe-Life Sciences Education,* 2015, vol. 14, no. 3.

[29] E. Riese, "Teaching Assistants' Experiences of Lab Sessions in Introductory Computer Science Courses," in *2018 IEEE Frontiers in Education Conference (FIE)*, 2018.

[30] F. J. Estrada and A. Tafliovich, "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,* 2017.

[31] V. Muzaka, "The niche of graduate teaching assistants (GTAs): Perceptions and reflections," *Teaching in Higher Education*, 2009, vol. 14, no. 1, pp. 1–12,

[32] V. Braun, and V. Clarke, "Using thematic analysis in psychology," *Qualitative research in psychology*, 2006, 3(2), pp. 77-101.