Individual Instructor Report Spring 2026 Version B for C S 371P - OBJECT-ORIENTED PROGRAMMING (53370) (Glenn Downing)

Project Title: **Course Evaluations Spring 2026**

Courses Audience: **70**
Responses Received: **66**
Response Ratio: **94.3 %**

Subject Details
merged_subject_id
merged_subject_display_name

Report Comments

Guide to the Interpretation of Course Evaluations at UT Austin

The goal of course evaluation process at UT Austin is to drive teaching excellence and to support continuous improvement in teaching and learning experiences. Course evaluations provide snapshots of student perspectives on their course-level learning experiences. Most experts on teaching evaluation advise that no individual method gives the complete picture of an instructor's teaching effectiveness, multiple and diverse measures, on multiple occasions, are advised to give a full picture of the teaching effectiveness of a particular instructor. Moreover, other factors, such as size of class, level of the class, and content of the course, can cause small variations in the ratings. Therefore, student perspectives for a particular instructor or course should be interpreted as a snapshot, and not as providing complete information on the teaching effectiveness of that instructor. For additional details, including the scales and how the Mean scores are calculated, please review the Report Guide at the end of this document or, UT Austin's Viewing Course Evaluation Results webpage.

Creation Date: **Friday, May 15, 2026**

Core Questions

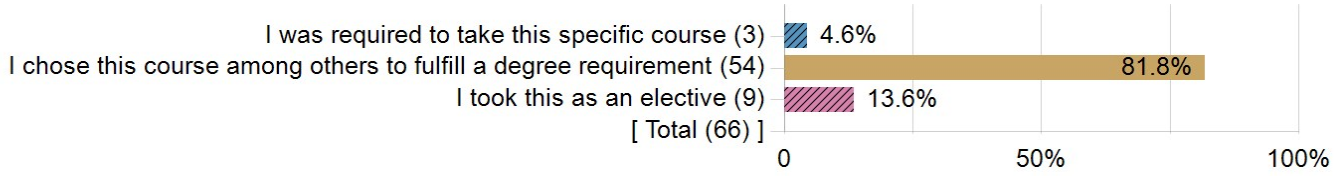
	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree	Responded	Mean
During this course, I gained a deeper understanding of the subject matter.	36.4%	56.1%	4.5%	3.0%	0.0%	66	4.26
The course was well organized.	54.5%	42.4%	3.0%	0.0%	0.0%	66	4.52
The instructor clearly explained the course objectives and expectations.	56.1%	39.4%	4.5%	0.0%	0.0%	66	4.52
The instructor fostered an inclusive learning environment.	54.5%	28.8%	10.6%	6.1%	0.0%	66	4.32
The instructor effectively explained the concepts and subject matter in this course.	48.5%	43.9%	6.1%	1.5%	0.0%	66	4.39
The instructional techniques kept me engaged in learning.	48.5%	34.8%	13.6%	1.5%	1.5%	66	4.27
The instructor checked for student understanding of the concepts presented in the course.	62.1%	34.8%	3.0%	0.0%	0.0%	66	4.59

Overall Questions

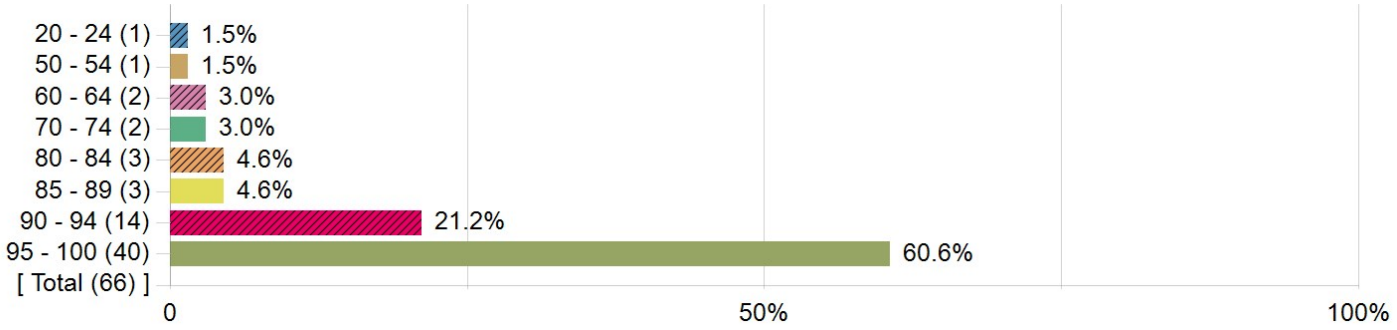
	Excellent	Very Good	Satisfactory	Unsatisfactory	Very Unsatisfactory	Responded	Mean
Overall, this instructor was	48.5%	34.8%	13.6%	3.0%	0.0%	66	4.29
Overall, this course was	39.4%	39.4%	18.2%	3.0%	0.0%	66	4.15

Other Course Questions

Why did you take this course?



Overall, approximately what percentage of the course meetings did you attend or complete (online, in person, or asynchronously)?



Statistics	Value
Mean	90.79

Comment Questions

Identify aspects of the course that were the most effective in helping your learning.

Comment
Cold calling and daily quizzes basically forced you to catch up if you missed anything. The quizzes being interactive was nice because you weren't just guaranteed a 0 if you happened to miss something.
The quizzes and exercises
Cold Calling
I thought the random calling, quizzes and exercises were a great way to keep students like myself engaged with the course material. I also enjoyed the papers as well, as some of the materials in the paper even came up in a later interview that I had which was super helpful.
The lecture style was engaging and helped me learn well. I liked that I didn't have to worry about multitasking to take notes while listening.
The course was well structured and Dr. Downing had a clear goal in mind when teaching, which helped in studying and understanding of concepts. Dr. Downing was also approachable whenever I had any questions.
Projects and Homework were helpful, as well as the step–by–step explanations of every operator / syntax feature of c++
I liked how we had a variety of activities to help us understand the material, like in–class quizzes, projects, exercises, and papers.
The active coversations
The Ed Lessons were helpful, along with the quizzes at the beginning of each class.
Cold calling, quizzes, ed lessons
The cold calling, it really makes you focus on what is taught. As well as the depth of the projects. It forces you to really pay attention to detail.
I would say the projects were pretty useful and so were the lectures
the projects and exercises
Active, present teacher. Cold calling. Constant energy.
I found the cold calling and the projects to help reinforce my learning and make sure that I am paying attention in class.
I found the quizzes and exercises to be very helpful in furthering my understanding. Having daily quizzes helped with recall of what we did previously, and having the quizzes be collaborative made it more enjoyable and also increased my confidence in answering them. The exercises, although sometimes stressful under time pressure, were still pretty fun. It was also super useful in implementing the ideas we had just learned in class that day as well as helping me build skills necessary for software engineering, like constantly testing.
Popcorn calling kept me attentive
The labs and his enthusiasm!
The projects
Surprisingly the cold calling helped me stay engaged as I would not want to be behind when called on.
Projects, exercises
The quizzes were good for making sure we understood the concepts.
I appreciated the LLM which was trained on the projects' data. UT Sage
The quizzes were the most effective in helping my learning as it required me to review the material before class.
[1] I liked how organized the course is. The schedule tab in the course website is very clean, clear, and consistent, and is always kept up to date. This means that we can always see what assignments have due in the coming days. Moreover, the structure of assignments is always consistent (i.e. blogs due on Sunday night, papers due on Sunday night –– it's always the same due date). This helped me to stay organized and understand what was happening. [2] Moreover, I found Dr Downing's approach of cold calling in class to be helpful. It's a good way to actually drive engagement and make sure that students are paying attention. It also doesn't crowd other students out, since Dr Downing is always willing and able to take questions about the subject matter. So, I enjoy that aspect of how Dr Downing runs class
The quizzes helped reinforce my knowledge in lecture materials.
The cold–calling helped me to pay closer attention to the lectures.
cold calling! helps a lot in identifying gaps
Exercises were helpful to reinforce the concepts I learned in lecture. I liked the last two projects, Darwin and Life, since they helped me learn how to write good object–oriented designs with no getters and setters. I wish the earlier projects also focused on those

Comment
skills more. I also enjoy Dr. Downing's lecture style and how he was clear and to the point with all the examples and motivation behind each example. It is a good way to get familiar with C++.
I found the daily quizzes to be a really good reminder of what we learned last class.
Exercises good at practicing the coding concepts we learn in class.
Downing is a wonderfully enthusiastic lecturer. The humor and passion for the subject matter was very engaging. The daily quizzes and cold calling were a bit intimidating but in the end I think these were very effective for reinforcing understanding.
I feel like the problems and the projects really helped me learn how to code better and be better at object oriented programming as a whole.
Doing the projects with a partner allowed me to learn many new concepts that I would have never tried.
The problems were good at developing my cpp abilities.
The projects were pretty effective in helping understand in class concepts.
I think cold calling people during lecture really forced you to want to learn or keep up with the material because you don't want to be wrong. And I don't think that pressure is bad, I think it is good because there are some times where I feel like I want to slack off with the material but then I realize that there is the possibility I will get called on during lecture so I want to make sure I know it.
Structure lectures
Quizzes
He makes sure everyone has an understanding of the material before moving on.
The lectures with cold calling were most effective in retaining information and staying engaged.
Cold calling kept me very engaged in the class.
I really enjoyed the problems we got for homework, because I feel that they mimick leetcode style questions which we need for interviews for internships so it's a valuable tool in the class. I also enjoy the quizzes being collaborative as they help me better understand the course material.
I liked the daily quizzes. They were a quick recap of the previous lesson. Being able to talk to others also reinforced my learning of the knowledge.
The weekly blogs were useful in supplementing and reviewing the material covered the previous week. Additionally, the projects, exercises, and daily quizzes were great ways to apply the concepts learned in class to actual applications.
The lectures were very helpful in terms of understanding the content. Knowing that I might be called on to answer a question kept me in focus.
Daily quizzes enabled material to be practiced several times rather than solely in one lecture.
I really liked that the class is active learning. Some people dont like that we get called on during class but this forces you to be focused, really try to understand, and stay on track during the whole semester.
The projects were laid out nice, and I found most of my learning came from working on them.
Cold calling was useful for class participation, and the exercises and quizzes acted as a way for us to learn while collaborating with one another.
Cold calling since it ensured I was paying attention to the lecture and was engaged throughout since there was a possibility of being called on. Exercises in class were also helpful to ensure we understood the content and could implement it ourselves.
Professor Downing is an excellent instructor. He's very passionate and well–prepared in his lessons.
Extremely well structured and organized with ease of access to necessary materials.
the projects and lectures
The daily quizzes and exercises were the most effective in helping my learning.
The lectures were very in detailed and the professor is very knowledgable on the subject

Identify the aspect of the course that you found most challenging, why you found it was challenging, and suggest one thing that could be done to help future students meet that challenge more effectively.

Comment
Strict attendance, got sick but still came to class to make sure I didn't miss an exercise. Missing a quiz is fine since there's a lot of them, but missing exercises isn't ideal. Maybe loosen up the policy for exercises.
The projects, it felt like there was a lot of fluff and busywork that wasn't necessary and actively slowed you down, with regards to many of the issues in the csv we got. Perhaps cut down on the side parts and focus more on the actual problem itself?
Exercises in class, sometimes there wasnt enough time.

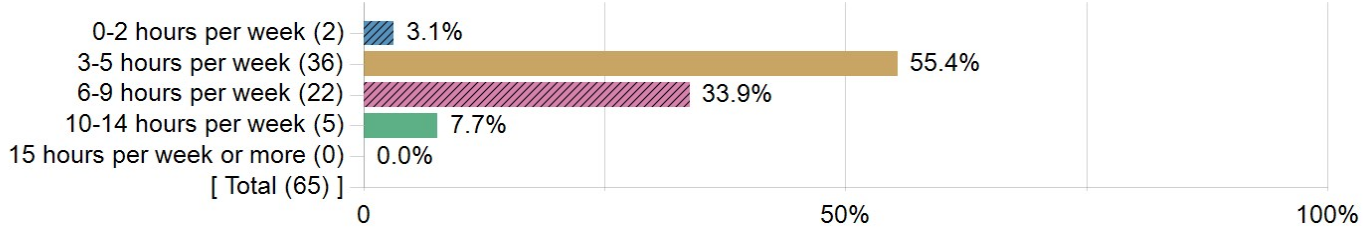
Comment
I found occasionally some of the problems on Kattis to be difficult and I felt that I spent a disproportionate amount of time on trying to solve a Kattis problem compared to a lab. For example, some Kattis problems were simple backtracking solutions whereas for others I somehow ended up in a probably incorrect 4d dp rabbithole.
The time constraints on the Ed Exercises were challenging. I understand why we had them but sometimes I would freeze and leave class feeling defeated.
The course can be time consuming, with several requirements that need to be fulfilled every week. This makes it difficult to manage with other classes.
I frequently felt frustrated about the mandatory attendance requirements. I commute, so it would've been nice if there was some sort of way to watch lectures asynchronously without being penalized.
The aspect of the course that was most challenging was probably understanding in the method of lecturing. I think it was a bit harder to learn in the format of constant questions rather than more explanatory ways. I understand that the cold calling increases engagement and is a way to check for understanding, but I would prefer if it was done after explanations rather than during.
Hard to follow some parts of the language, but those just come with practice.
Showing up on time
The final two projects were extremely time consuming, but not because of complexity, and instead of the number of required steps that felt useless and in personal experience not useful in industry.
none, I feel like this course was very well organized and expectations were clear throughout
Learning the new syntax. No real critique, it's just par for the course.
The hardest thing was the excercises. I felt like sometimes I didn't have enough time to finish them.
the quizzes and exercises were often down to the wire on time, I think that was by design but an extra minute or two would have encouraged more thinking on my part and less reliance on collaboration
Most challenging – c++ specifics like references and pointers
I found the exercises in class to be challenging because the 20 minute time to complete it was a bit stressful, and sometimes the code that we had to produce was a bit daunting. One thing that future students could do to prepare for this is to pay close attention in class because a lot of what is in the exercise is talked about minutes before.
I found the projects and the problems to be challenging. The problems weren't very related to the course content, rather mostly focusing on Leetcode–esque skills like dynamic programming. The projects were a little confusing, especially Darwin since we weren't given a day in class discussing the project content until weeks after the project was published.
Embarrasment from not knowing the answers in popcorn calling
N/A
Assignments are definitely challenging.
excercises bc of time limits – maybe walk through them in the last ten minutes instead of just leaving us to solve them
The time limits on exercise would be cut down when the lesson went long, making it harder to complete.
I found the exercises the most challenging, because they would be in the topic that we discussed in class, but the class notes are only published after class. I would have appreciated if the professor made the notes available immediately on the days of the exercises.
I also found the quizzes to be the most challenging aspect as well. Some days the quizzes would be difficult with several similar answer choices.
[1] Personally, I found it hard to not be able to take notes in class. Dr Downing's structure is that the professor takes notes and publishes them at the end of class. While this is very nice, I personally prefer to be keeping track of the material on my own. It's a good way for me to ensure that I'm actually paying attention, since I need to be on top of my own record of the topic at hand. I found it easy for my attention to drift during class since I couldn't take my own notes. [2] Secondly, I feel like the subject matter of the course could be structured differently. For the first few weeks of the semester, it felt like we were primarily just learning C++ syntax instead of actual object oriented design principles. Though I found that interesting, I feel like the actual objective of the course with object oriented design was primarily learned through assignments like the papers, and not through actual in class discussion. So, I think that in class lectures talking about some of the content from the papers could be helpful. Moreover, a restructuring of the course to talk about classes earlier in the semester would be nice, if it's feasible to do so.
The most challenging part of the course was starting the project and determining what to do first. However the in–class description of the project helped with this.
The exercises and the quizzes are difficult, especially during time crunches. To succeed, you need to read each line from the beginning carefully. If you panic and try to find shortcuts to save time, it usually doesn't work.
the tests

Comment
I wish Dr. Downing would look out for questions from other students more rather than just the student he was cold–calling. I also wish we stayed on schedule as the topics near the end of the syllabus (smart pointer, unique pointer, modern C++ features) were the ones I was most excited about learning and I wish we had more time in the semester for those topics. We could spend less time on the introductory stuff to allow time for those topics, since the content became increasingly interesting later on in the semester.
I think the hardest part was remembering stuff we learned long term. The quizzes helped remind us what we learned the class before, but it was hard to retain the information later on.
Cold calling made be very anxious for every single class.
Occasionally the makefiles had issues that needed to be corrected before code would compile correctly. This is a very minor issue though.
I found the quizzes dificult but that's because I wasn't studying as well for them. The notes from last time I definitely could have read better and thought about more and it would've been better.
Some of the in–class quizzes seemed difficult due to the very short time constraint.
The projects are so time consuming and it is dismotivating when you have to do so much bulk work that do not affect my codes output the papers also seem outdated for modern software development.
Daily quizzes were challenging since it required that we study before each and every class, and with more and more classes at UT using this format managing my time seems to leave my own control.
I think the papers felt very repetitive and while I do like reading about it and understanding what is going on, there where some times where I felt it was boring and felt even more disconnected with having to write comments. While my suggestion is really broad, I think there is a better way to do readings or papers than the Perusall and writing comments.
N/A
Projects. More info on projects
I found the time constraint on some of the exercises to be challenging.
The most challenging part of the course are the quizzes. To do better on the quizzes, more time allotted for the quizzes would help.
I found the exercises (ed lessons) the most challenging because of the time constraint. I would suggest to first figure out how the problem should be solved before typing any code.
Certain concepts in the class were difficult for me to grasp as C++ is a pretty complex language in my opinion. References, pointers, const pointers, etc. can get confusing at times so maybe certain difficult topics get more in depth review.
I found the projects most challenging, especially the first one. I say this because there were no instructions. All there was is a hackerrank and a csv file that tells you the issues. But it doesn't really tell you how you should approach the project. There was also one project where you were expected to start from scratch, and that was very confusing as well. I think that more instructions should be provided in the future.
The most challenging aspect of the course was the projects, as they required lots of careful planning of the solution architecture to remain in accordance with the specifications. I don't have much feedback to help future students meet that challenge, as I believe that although challenging, it was still a great experience working through the project process.
I found the quizzes to be the most challenging because of the time limit. That pressure to submit in time created an environment where I couldn't think coherently. I was more focused on making sure that I gave each question enough time and then I couldn't think critically. I would suggest removing the time limit and simply waiting for all students to be done with the quiz.
Cold calling often leaves some students confused as the direction the professor tries to push them is unclear.
I really struggled a lot with the assignments at first. Having to do all the issues and many little things was very hard and challenging but it really helps understand how real life projects work.
I found the format of the class to be a struggle. The format was to have the professor briefly go over the material, and then cold call on students. Sometimes he would cold call prior to explaining a topic. I found this difficult because it was a challenge to hear my peers try to answer, or they would provide inadequate answers, which led to me being confused. The cold calling also made me anxious to come to class, and not excited.
Sometimes it would be easy to to get lost during lectures, as concepts build upon one another and are sometimes covered very fast. I think the way the typed notes were done was also a little hard to follow, and something more structured might help.
I found the exercises most challenging since it was hard to think through the new concepts under time pressure. Maybe giving a heads up that we would have an exercise in lecture to encourage more students to study the last lecture notes prior to implementation, although I know the point of exercises is so that they are surprises.
N/A
Most challenging was the tedious nature of the volume of assignments.
the projects were challenging maybe a better spec

Comment
I found the exercises the most challenging due to their time limit. I think having the TAs more readily available to help during exercises would help future students meet that challenge more effectively.
The quizzes and exercises were quite hard and many times they include intuition not yet covered in class

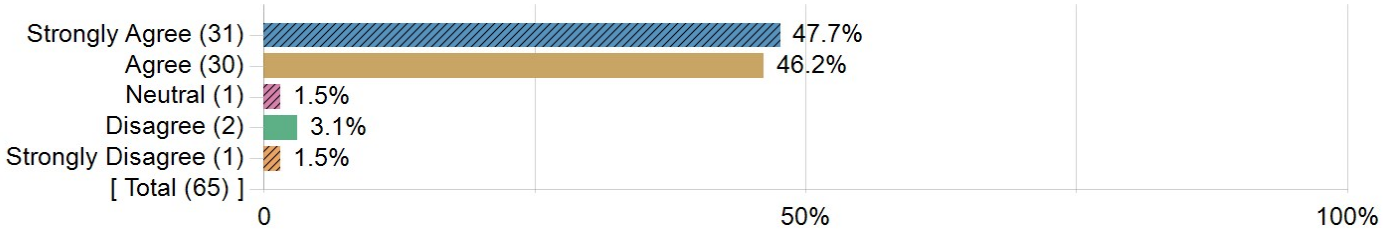
College, School, or Unit Questions

On average, approximately how many hours per week did you spend working outside of the course? Include time on homework, reading, reviewing, papers, projects, etc.



Statistics	Value
Mean	2.46

The course format (online, hybrid, face-to-face) helped me to learn.



Statistics	Value
Mean	4.35

Report Guide

Guide to the Interpretation of Course Evaluations at UT Austin
The goal of course evaluation process at UT Austin is to drive teaching excellence and to support continuous improvement in teaching and learning experiences. The two sets of scales used for core evaluation questions and the associated weights are:

Strongly Agree (5)
Agree (4)
Neutral (3)
Disagree (2)
Strongly Disagree (1)

Excellent (5)
Very Good (4)
Satisfactory (3)
Unsatisfactory (2)
Very Unsatisfactory (1)

The Mean is calculated by adding all of the weights for a single question and dividing by the number of respondents. The course workload question is not averaged.

The number of students (e.g. respondents) marking each option is reported for each of the items. These frequency distributions provide information about the level of student ratings and the spread and shape of the class distribution of responses. The distributions thus provide a picture of student perception of a course.

Course evaluations provide snapshots of student perspectives on their course-level learning experiences. Most experts on teaching evaluation advise that no individual method gives the complete picture of an instructor's teaching effectiveness; multiple and diverse measures, on multiple occasions, are advised to give a full picture of the teaching effectiveness of a particular instructor. Moreover, other factors, such as size of class, level of the class, and content of the course, can cause small variations in the ratings. Therefore, student perspectives for a particular instructor or course should be interpreted as a snapshot, and not as providing complete information on the teaching effectiveness of that instructor.