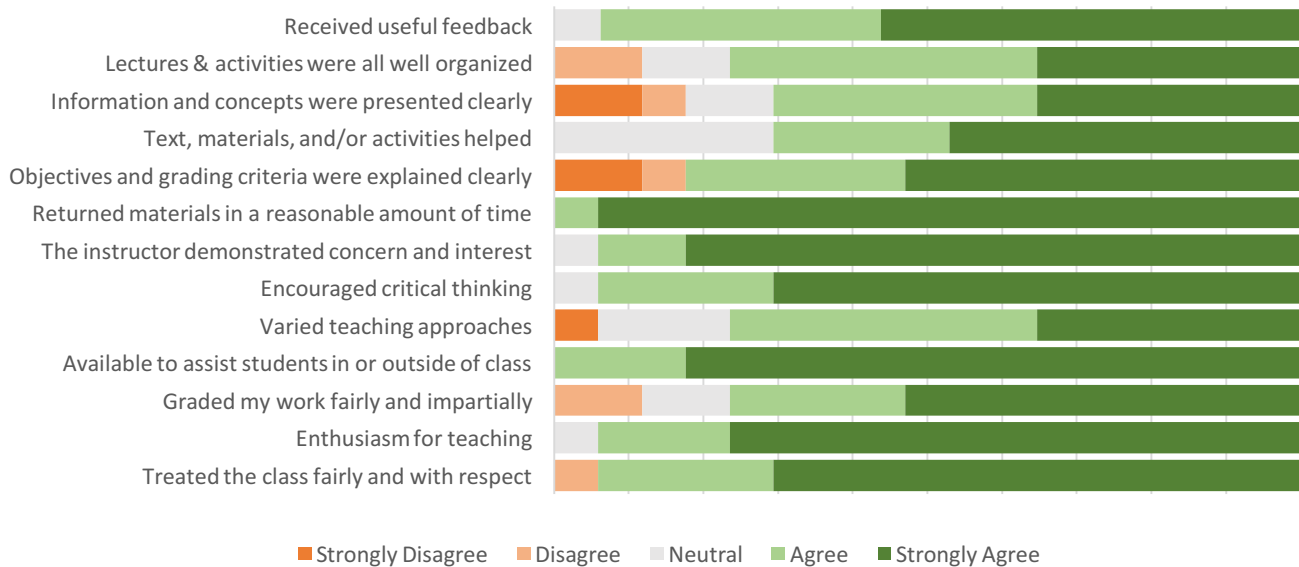


# CSCI 330 – Survey #1 Responses (n = 17)



## List some things you like about the course

- I like how when we missed a class you made a video instead of throwing twice as much at us in the next class. I am absolutely loving this course! It's really changing the way in which I look at things and making me a much better computer science student overall. I really appreciate all the hard work you do and your willingness to help students even if they aren't your students. You are the best teacher I've had and the reason I didn't transfer to a different School.
- Generous feedback on D2L - Fair grading
- Um, I enjoy thinking of programming in a way that "maps" to mathematical functions far more directly than imperative programming. I definitely appreciate Killian's commitment to rapid turnaround times, and his expressed attitude of being willing to help students. And, like, nothing's been particularly confusing so far.
- I like how there are usually multiple examples for a problem. It helps to get a concept if shown from multiple views.
- I like that we're learning about both OCaml and how/why programming languages work, and it's interesting learning about a functional programming language and how different it is from Java and C++. The many examples gone over in class help a lot in understanding it too.
- The challenge, OCaml, the labs, the topics we have covered so far
- The instructor's attitude to the class and as to how he interacts with his students. He almost seems to come off as relatable to the students. He also almost seems to bridge the gap between teacher and student just by his personality and how he interacts with his students. This is a good thing and not at all an insult just in case if there was any debate behind this.
- Professor is upbeat and makes learning things not so dull, material taught is really interesting, it is really cool learning to code in another language.
- The instructor does a great job teaching the class. If it wasn't for him, I doubt I'd enjoy the class at all. Maybe the next units will be better, but this first couple would have been the death of me if not for our professor.
- Very enthusiastic always in a good mood always willing to help students stays on campus a ridiculous amount of time to help anyone who asks
- I like how available the instructor is outside of class for questions and help  
The prof's enthusiasm How the prof interacts with students
- Really challenges one to think.
- NONE

## List some things you don't like about the course

- I feel this instructor grades a little too harshly in some cases. There have been some times where it seemed it was just the smallest detail/part that was wrong and the amount of points taken off could seem a bit much for such error. Keep in mind that this isn't a case where such professor is dropping 0s and Fs for missing one detail. It's not like that at all. I am just stating that sometimes the instructor's grading can be a bit too harsh in some cases but other than that everything else is good. *[What is too harsh? I would argue that it's fair grading. If you answer a question mostly correct, I will award most of the points for the problem. If you make a claim but don't support it, I can't award full credit]*
- I wish we could learn more about visual studio or the development platform of choice. *[This is out of scope for the objectives of the course]* When prior knowledge is similar to something we're learning it'd be nice to give a few reminders to check ourselves before we wreck ourselves. For example,  $a = b + 2$  is an assignment not an equation like in mathematics. *[We are in the context of computer science. When we have code samples in C-like languages, we know = is assignment]*
- Material learned is sometimes tough to understand and I think that more examples of some things could really help students understand it better. *[I agree; I'm working on better examples]* Other than that, no complaints.
- OCaml (I enjoy the concept of functional programming, but there are better alternatives that are used much more frequently in the industry) - 8 AM lab *[The course is not about learning a practical, industry-used language. Once you know a functional language, you can pick up any language]*
- The exam questions are vague. *[If any question was vague, you could have asked for clarification during the exam period]* The exam questions are deceptive. *[How?]* The exam questions reverse examples seen. *[There was no test question where I reversed what was expected of you]* One exam question had never been shown to us prior: weakest condition. *[I went over weakest precondition on the homework and in lecture. I also went over the homework in class]* The exam grading criteria is unknown. *[When I went over the exam, I explicitly provided grading criteria for each question]* The exam grading criteria is subjective. *[No, it was not; If you want to contest any part of your exam, you have until Friday, March 2<sup>nd</sup> at 5PM to do so]* The study guide served little purpose. *[The study guide clearly listed everything on the exam. Every question mapped directly to one or more component on the review sheet]*
- Homework and exam were harder than what was explained in class *[Yes, it was; I also have office hours and can answer any questions you have over email or Piazza. I am always in the department and try to go out of my way to make sure I am available to help any students].* lectures were explained using very easy examples. And, exam contained the harder versions of those topics. eg. WEAKEST PRECONDITIONS *[The example on the exam was easier than any example we went over]* When someone asked to do an example of loop invariant before the test, he said it won't be on the test (atleast not the proof and stuff). But, it was. *[The proof and "stuff" was not on the exam. I asked you to define loop invariant and identify where the loop invariant must hold]* If we are confused about when to check conditions then how can we draw arrows showing that the condition is true there? *[If you were confused during the exam, you could have asked for clarification. I explicitly mentioned that a loop invariant proof wouldn't be on the exam. That doesn't mean that nothing about loop invariants would be asked on the exam]*
- Sometimes burns up my brain.
- I don't like autolab *[Sometimes, me too 😊]*
- Sometimes get off topic
- Unit 3 is the worst.
- At this point nothing
- N/A
- N/A

## What could you change, if anything, about the lectures?

- I wish we could do more in class work or maybe have an hour or two a week where students and the professor could meet up and walk through some extra problems or watch how a mock lab assignment is broken down and the lab is coded from scratch. Videos would be awesome too and set up a way to comment on them if we have any questions. *[It's a lot of extra effort and planning to do this. Part of the labs are for self-exploratory and learning]*

- I would add more concrete examples, because of the abstract nature of the subject matter. Many students have difficulty learning general concepts and need more concise applications of the topics. *[Concrete examples are harder to realize for abstract nature of subject material. Future concepts will be more concrete]*
- There are already usually a lot of examples, but I think it would help to have even more examples. *[Future components of the course fair better to additional examples. They are coming]*
- Maybe just a bit more information on the slides, mainly when code comes into play. *[The slides are super dense; I'll have more material referencing code later on]*
- I would like them to emphasize specifically what we need to know for the exam. For example, "Given a question like [this], you should answer like [that]." *[I will never teach to an exam]*
- I don't understand math notation at all. He teaches it fine as it is, but I wish there was a better way to represent these concepts. *[What specifically about math notation?]*
- talk a little bit more about OCaml in class, almost everyone seemed to have a very hard time with the most recent lab, and most of it was people just people not understanding the syntax and concepts of OCaml *[the second lab had no new concepts of OCaml which is why I didn't cover anything new]*
- need more examples of ocaml to understand lab assignments *[as new concepts are introduced there will be more course time spent covering additional concepts]*
- He should start doing some real-life problems (related to HW and exam) on the board rather than coming up with some random stuff. That confuses me. *[The random, simple examples are designed to get you to understand the problems at a fundamental level. Once you understand a concept, you can then apply it to homework and exams]*
- I don't think that I would I change anything about the course thus far.
- Can't think of anything off the top of my head.
- At this point, nothing
- N/A

## List any additional comments

- I feel like a fish that's learning to climb a tree here at school. But on the flip side when I'm back into my element I'm even better than I was before. I just there were more hands on learning opportunities. *[330 is a theory-dense course with lots of abstract information. Hands-on comes mostly with the labs. I can try to have additional information in future lectures]*
- Please post practice exams with answers worth full credit. *[No. I will not do this]* Those that did well on the exam are as confused about the scores as those who did poorly. *[This is a subjective statement and I have received no comments from students supporting this claim]* Your assessment style leaves us guessing at what answers you expect, not the material learned. *[If you have a problem, you need to see me. Typing this in an anonymous survey does not solve anything as it only makes me aware of how someone feels. If you feel like I have treated you unfairly, you need to see me. The last day to contest anything with the exam is Friday, March 2nd at 5:00PM]*
- -topics were not clearly explained. ex. weakest precondition, attribute grammars, loop invariant. *[I went over everything I expected students to know about these topics. The homework included questions related to weakest preconditions and attribute grammars. I also went over complete and full answers for the homework]* -If there are many different things about a topic, then we need to know all of them in order to do good on the test. we were told one thing and were asked about a completely different thing on the test. *[Without clearly listing what you feel I unfairly asked, I have no way to address this comment]*
- Keep making us laugh and make learning fun, I think you're doing a pretty good job as a professor.
- The best professor I have had at explaining concepts and helping with things I do not understand.
- The professor is always trying to help students in any way he can
- Great class and instructor - Cares about teaching and students.
- You're a great professor, keep going!
- Keep up the good work
- OCaml is surprisingly enjoyable.
- Got nothing
- n/a
- N/A