

# Writing Reviews

By Example

# Less Helpful Reviews

- had fun

- you were just alright.

- He is a funny instructor, and i enjoy taking this class

- Wowwww during my time spent in this class I actually think I un-learned how to properly program. Can we learn useful stuff if we're paying 5,000 a quarter to earn a CS degree.

More Helpful Reviews

- I hope the instructor can white board out stuff more



- I like how you replaced the final with a project. Aside from the iterator lab, which took a few hours longer than it should have, the labs were well written and fairly straight-forward.

- Using more visuals during lecture (e.g. UML diagrams) and explaining code using the visual would be helpful, but otherwise, the material was well taught.

- It would have been better to work more on design patterns during the labs, as they're the most complex thing seen in this class. Reading the examples in the slides was enough to understand the principle, but not enough to be able to use the design patterns by ourselves.

# Ideal Reviews

- The instructor was very open to any questions we had, the problem was that we did not receive enough of a grasp on the present material in order to ask good questions when we had the opportunity to do so. The language used during much of his lectures was very technical and often hard to understand and follow a long from a non-professional/undergraduate student perspective. Computer Science is technical in nature, but he could have given more effort to communicate effectively to the audience by break the ideas down into a more consumable method. The UML diagrams and Glyph examples he kept referring back to often would not help so much. The hand-written code examples helped a lot though so I wish he would have done more of those on the board instead of just reading from the powerpoints. I would have also benefited if we would have compared and contrasted the design patterns more often to really understand their purposes and differences in a practical manner. You can only be orating so much in a CS lecture before you start losing your effectiveness in communicating the material. We would have benefited so much more to see live coding examples of the design patterns! And I understand that time was an issue since the lectures were very short so the opportunity for live coding probably wasn't there, but pre-recorded videos could have replaced live coding examples instead. We need to see the action! Like in a math class were you see an example fully worked out. Also the homework assignments involved way much more work than we were prepared for in lectures, and their instructions needed more clarity. Also, it was extremely hard to study or know how to properly prepare for the material on the exams given the level of difficulty of the problems, the short amount of time we were given and not being able to look something up when you need to program something, it goes against the nature of programming and its just not a practical method of assessment/evaluation for CS students to code by hand, or even by memory.