

CS 280 - Code Reviews

One of the goals of this course is to improve your ability to read and review existing code. For each submission, you will be assigned another random student in the course. You must review their code and offer constructive feedback. In industry, this process is known as a *code review* and is frequently used to improve software quality and to catch software defects early.

This document lists the questions about the code that you should answer for each submission, giving examples of both *non-constructive* and *constructive* feedback (aim for the latter!) for each question. Each review should be at least 2-3 paragraphs, and you should focus on deeper criticisms (algorithm/data structure choice, algorithmic complexity, etc.) rather than more superficial criticisms (code style, formatting, etc.).

Question	Example of Non-constructive Feedback ✘	Example of Constructive Feedback ✔
<i>What did you like about this submission?</i>	"It was good." "It was clean."	"The main loop in function X was clean and well-documented." "I liked the use of extra indentation to line up the array initializations in function X."
<i>Describe one significant difference between your own submission and this person's submission. Which approach is cleaner? Which approach is more efficient?</i>	"My code is faster." "My function X is shorter than this author's version. Mine is better."	"I chose to calculate the maximum value on every iteration of the outer loop while this author calculated it only once and cached the result. Their approach is more efficient, but my approach works even if the list is modified during iteration."
<i>Is the code well-formatted and well-documented? If not, suggest some specific improvements.</i>	"Not enough documentation." "I couldn't understand the code."	"The goal of function X was unclear; the author should add some documentation regarding its inputs and outputs." "The code is inconsistently formatted; the author should consider removing the extra empty lines in functions X and Y."
<i>Did you find any software defects? If so, briefly describe them.</i>	"Function X doesn't work." "I couldn't get it to run."	"Function X does not produce the correct output for this input: 'ABC'" "The program crashed with an IOException when I tried to run it on inputs with more than 4K lines."
<i>Do you have any other constructive comments for the author?</i>	"This code sucks and needs to be rewritten." "This code is perfect."	"Function X has redundant if-conditions; the last two could be consolidated." "The use of recursion in function Y to avoid ugly class-level data structures is very elegant."