

Program Grading

Your programs in CS106X this quarter will be graded on two major criteria: functionality and style. Although you will often concentrate your efforts on getting the program "to work" (whatever that takes!) you'll learn that our expectations go beyond just "working"— we also want you to strive to develop a style for producing elegant, well-engineered solutions. A program can truly be a work of art and something you can be proud of when showing off both the executable and the code.

Functionality Functionality denotes whether or not the program performed as we expected i.e. did it meet all the requirements set out in the assignment handout. For this evaluation, we examine your program's behavior from an external perspective, without looking at the code. We will test your program on several different inputs to determine its effectiveness in a variety of situations.

Style/design Style captures whether the program is designed in a straightforward and clear way that is easy to read and understand. Does it employ principles of good decomposition (how the program is divided into procedures) and re-use rather than repeat code? Does it make well-reasoned choices for constructs (for example, `for` vs. `while`) and data structures? Would it be easy to make modifications? Are the variables and procedures logically named? Does it have descriptive and accurate comments? Are the layout choices effective and consistent (use of white space, capitalization, curly brace placement, etc.)? For this evaluation, we carefully read the code and offer constructive input on your choices.

Each category is assigned a grade using a bucket scale (the two different scales used are described below). Rather than resorting to nit-picking and summing up points, we want to provide a clear overall sense of your program's strengths and weaknesses in these areas, and we believe a fairly coarse bucket scale is best for this. The total assignment score is the sum of the scores, the two parts are equally weighted. It is possible for a perfectly functional program to earn a low score in style and vice versa. We encourage you to make your program shine in both areas!

Interactive grading

All the programs (save the final assignment) will be graded in a one-on-one conference with your section leader. Past 106 students seem in strong agreement that this interactive grading is one of the course's best features. Your section leader will help you to see where your program excels and where you can improve for next time. Their feedback and nourishment can help guide you in your passage from programming novice to accomplished software engineer. The grading conference is a marvelous and rare opportunity for intense, personal feedback. I hope you'll find it a lot more useful and inspiring than getting a returned paper marked with a numeric tally.

Functionality scale

Below is an explanation of the scale we will be using to express the level of functionality in a given program.

- + An exceptional program—one that works correctly in all situations and exceeds our expectations, going above and beyond the assignment requirements. Super work! We cannot find fault and would love to show such a program to the entire class as an extraordinary example. Only around 5% of the programs turned in will receive this grade and it is never given out unless a ✓+ was also received for style.
- ✓+ An excellent program— one that meets all of the requirements gracefully and has only minor quirks. Your section leader may be able to provide suggestions for some small adjustments or quibble with a few of your choices, but these programs are overall very well-done and distinguish themselves as an excellent result. Perhaps a third of the programs will receive this grade.
- ✓ A good, solid program— one that is complete and meets the requirements of the assignment, although perhaps some parts could have been handled better. Your section leader can provide feedback on what can be improved upon for next time. This is the median grade and we expect around half of the programs will be given this grade.
- ✓- A program with room for improvement. The program doesn't quite meet all of the requirements of the assignment, perhaps because it is a bit incomplete or has errors with one or more required components. With a bit more work it could be a good program and your section leader can help you to understand where to go from here.
- A troubled program. The program exhibits serious problems and/or is an incomplete attempt, but it does show some understanding and effort. This grade indicates that improvement needs to be made immediately.
- A very incomplete program. This score usually indicates the program had very little effort put into it and/or has serious errors. A large amount of work needs to be done before this program would be complete.
- 0 No program turned in at all.

On some assignments there will be an opportunity for extensions beyond the standard assignment requirements. Doing this extra credit has the possibility of turning an excellent grade (a ✓+) into an exceptional grade (+). However, extra credit will not make up for a problematic assignment and will not affect any program that would normally receive a grade of ✓ or lower in either functionality or style. Thus make sure you have thoroughly tested your assignment core before moving on to extra credit!

Style scale

There are fewer buckets in the style scale and we expect the style scores to be a bit higher than functionality. Below are the buckets used for style evaluation:

- ✓+ A well-engineered and readable program. Bravo! The overall approach is straight-forwarded, the tasks are nicely decomposed, the algorithms are clear and easy to follow, comments are included, the layout is consistent— an overall polished production. All students are capable of achieving this result and we encourage you to set your

goal high. We would love to give out half or more style grades in this bucket.

- ✓ A program that exhibits reasonably good style. The program clearly makes a sincere effort toward an elegant solution, but it may not be entirely consistent, have a few minor problems or perhaps a more major issue that needs attention. Your section leader will be able to offer constructive ideas on how to tidy up and refine your work.
- ✓- A problematic program. The program has noticeable flaws, such as a convoluted approach, poor decomposition, overall sloppiness, or disregard for stylistic concerns. Often this grade means that not enough effort or attention was given to this area or that the student needs more coaching about what is expected. Certain programs, such as those that are very incomplete, will automatically be placed in this bucket.
- 0 No program turned in at all.

Lateness

After your self-granted extensions have been exhausted, late programs are penalized one category per class day late. This is a strong incentive for you to plan your schedule to meet the deadlines and conserve your valuable late days for true necessity. Since the programs are assigned one after another, it is also important to consider the impact of late work on the next assignment. Don't let the "domino effect" bring your entire quarter down!