Programming style

Kevin Schmidt, Susan Lindsey, Charlie Dey

Spring 2019

Programming: you, your program, your grade

- It's a necessary condition for your program to compute the right thing.
- But programming style is also important: we may subtract points for an 'ugly' program.
- We will read your source. Make sure it is written in such a way that we can understand it.
- It's all for your own good: badly written code is hard to debug, hard to understand for your colleagues or even yourself half a year from now.

As Donald "I am not worthy" Knuth puts it

The best programs are written so that computing machines can perform them quickly and so that human beings can understand them clearly. A programmer is ideally an essayist who works with traditional aesthetic and literary forms as well as mathematical concepts, to communicate the way that an algorithm works and to convince a reader that the results will be correct.

Programming languages are about ideas

A powerful programming language serves as a framework within which we organize our ideas. Every programming language has three mechanisms for accomplishing this:

- *primitive expressions*
- *means of combination*
- *means of abstraction*

Abelson and Sussman, The Structure and Interpretation of Computer Programs

Abstraction is good

Abstraction means your program talks about your application concepts, rather than about numbers and characters and such.

Your program should read like a story about your application; not about bits and bytes.

Good programming style makes code intelligibe and maintainable.

(Bad programming style may lead to lower grade.)

Language features

Just because a language has a certain feature, does not mean you need to use it.

- Being too clever may give hard-to-read code. May even make your code slower.
- Some C++ features are really from C: no longer needed.
- The four-letter word that starts with g should never be used.

About using the internet

- Yes, you can find solutions on the internet.
- No, that will not make you understand what you're doing.
- We are giving you a sequence of building blocks: try not to use things that you haven't been taught.