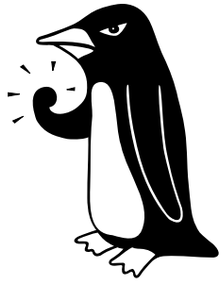
Writing Good Software



“The Angry Penguin“, used under creative commons licence
from Swantje Hess and Jannis Pohlmann.

Warwick RSE

Principles

Rules

- Write code that works
- Follow Language Standards (E.g. Fortran 95 or 03)
- Turn on compiler warnings and fix them wherever possible
- Use version control
- Make things easy on yourself - l v.s. 1 vs I, Name v.s. name v.s. names
- Write code you can verify
  - Modular structure
  - Test problems
  - Write a bit, check or test it, move on

Kinda Rules

Things that aren't actually hard rules, just usually a good idea

- Plan before starting, prototype difficult or unfamiliar bits
- Keep functions reasonably sized- do one thing and one thing only
- Avoid global variables that aren't vital global state
- Don't over-complicate until its needed
- Don't repeat yourself unnecessarily - write a function instead?

Inline Comments

- Comments best used to:
  - Cite source of algorithm, give human readable equations
  - Note limitations of algorithm or method near where its used - you'll need to know this if it changes
  - Explain things that aren't clear from the code itself - but see Self-Documenting code later
  - Can use for TODO's etc also

Documenting Code

- Document what things do, not details of how coded
  - can be invalidated if code changes
  - can always be inferred from the code - redundant information (see also Git later)
  - Say what parameters do, what is modified and what is returned
  - Explain limitations, prerequisites, invariants
  - Also consider user documentation - e.g. walk through a simple test problem

Self-Documenting Code

- **Not a substitute for documentation**
- BUT you can name variables, functions etc so as to indicate what they do
- E.g. constants. 86400 v.s. seconds_in_day
- Don’t go overboard - max 5-ish words
- Avoid being too specific - don’t want to have to rename a function/variable if irrelevant details change

More Resources

- https://warwick.ac.uk/rse/training/
- https://warwick.ac.uk/rse/training/introdev
- http://www.stack.nl/~dimitri/doxygen/
- http://www.fortran90.org/index.html
- http://fortranwiki.org/fortran/show/Tutorials
- https://stackoverflow.com/