Tutorial Rules of Thumb

Bjarne Stroustrup

Mostly hint meant for reading;
not organized as a talk

Tutorial rules of thumb

- Introduce your topic
  - What are you trying to teach people?
  - What benefits might a reader get?
- Try to imagine what your reader is thinking when reading
  - And what your reader needs to know (at first)
  - Decide between casual and professional style (“professional” is the default)
  - Possibly state your assumptions about the reader explicitly
- Give some idea of why something is the way it is
- Give some idea why this is worth reading
- Keep the introduction – first page – brief and light

Tutorial rules of thumb

- Go from the very simple to the detailed
  - “Hello, world!”
  - Re-read the K&R introduction
- Base your presentation on code examples
  - Show both definition and use
- Use graphical representation where possible
- Tables can be useful for summarizing
  - graphs are usually better still
- Don’t turn a tutorial into a manual

Tutorial

- Define your terms (informally)
- Use an active voice
- Address the reader directly?
  - “You can open the can like this”
  - “We add two vectors like this” Compile and run your code examples
- Keep sentences (and thoughts) simple
- Have a native English speaker look over your text
  - At least use a spell and grammar checker
  - Give references (“more information”)

Tutorial

- Use different fonts for text and code
- Lay out your code for readability
- Use reasonably realistic examples
  - No “f(x)” or “cuddly animal” examples
  - Finding good examples is often the hardest part of writing a good tutorial
  - If possibly make the examples a series of related examples building up to something useful/realistic
- Keep examples short
  - Eliminate distracting details (even if those are necessary at scale)

Tutorial

- Be fair
  - Don’t oversell
  - Don’t fail to mention significant problems
  - Don’t lie (e.g. by making a general statement where major limitations exists, such as “all objects can be assigned to”)
- Sometimes it is a good idea to show and warn about common errors
- Use a consistent style
  - Layout, phrasing, degree of detail
- Go read the tutorial from K&R (again)
  - Make it look simple, easy, and obvious
  - Make it useful
  - Don’t try to look clever
- Practice, re-write, get comments from “friendly” readers before shipping

Tutorial

- Always introduce code with text
  - saying what the code is supposed to do
  - Don’t use page-long examples
- Always illustrate major points with code
- Don’t say anything unless you have a point to make
  - Simply listing code without comments is not helpful
- Start with a “Hello, world!” example
- Say what you are going to say; say it; say what you just said
  - Introduction; body; summary – apply recursively, if needed
- Structure the tutorial with named subsections
  - Don’t write a rambling narrative; this is ***technical*** writing
- Take great care about the first few paragraphs
  - If you get those wrong (boring, uninteresting, uninformative, too cute) the reader will read no further

Tutorial

- Be concrete
  - Go from the concrete (e.g., vector or sort) to the abstract (e.g., container or algorithm)
  - Use more than one concrete example
- Don’t write “bottom up”
  - Show benefits and compelling examples early
  - Don’t present features in historical order
- What can you say that improves on what is already available?
  - Use multiple sources
- List your sources
  - Not just at the end: embed references in text to show what comes from where
- Humor doesn’t travel well ☹
- Don’t assume that the reader has color
  - E.g., color graphs are unreadable when printed in black and white
- Beware of material that dates a text

Tutorial

- Ask for feedback
- …
- This is not at all easy
  - “do what I say, not what I do”