Engineering Methodology

- Requirements
- Algorithm
- Example
- Coding
- Debugging
Figure out what problem you are solving.
Read the specification very carefully.
Find out what the customer really wants.
Understand the domain of the problem.
Become familiar with the state of the art.
Evaluate what the competition is doing.

Make sure you are doing the right thing!
Algorithms

- Figure out how to solve the problem!
- Consider existing solutions if available.
- Evaluate several different approaches.
- Take into account cost, performance, etc.
- Review whether you are meeting requirements.
- Do not optimize prematurely.

Avoid coding!
Example

- Document one or more examples of the algorithm.
- Start with the centerline case.
- Add corner cases and error handling later.
- Use representative data for the domain.
- Create a detailed reference for later stages.
- May be revisited during coding and debug.

Someone else should be able to read it!
Coding

- Translate algorithms into code.
- Start with an architectural block diagram.
- Learn the programming language inside out.
- Understand the operating system environment.
- Become an expert at tools.
- Understand the underlying hardware.

Incremental development rules!
Testing and Debugging

- Test individual methods before using them.
- Not just centerline, also boundary conditions.
- Debug your code in a methodical fashion.
- Thoroughly investigate any anomalies in behavior.
- Make sure to test all the branches in your code.
- Anything can be debugged, given enough time.

If you haven’t tested it, it doesn’t work!