Software engineering

The collection of processes, methods, techniques, tools and languages for developing quality operational software.

External quality factors

- **CORRECTNESS**
- **ROBUSTNESS**
- **INTEGRITY**
- **EASE OF USE**
- **REUSABILITY**
- **EXTENDIBILITY**
- **PORTABILITY**
- **EFFICIENCY**
- ...

**Correctness:**
- The ability of a software system to perform according to specification, in cases defined by the specification.

**Robustness:**
- The ability of a software system to react in a reasonable manner to cases not covered by the specification.

Software quality factors

Product quality (immediate):
- Correctness
- Robustness
- Integrity
- Ease of use
- Ease of learning

Process quality:
- Timeliness
- Cost-effectiveness

Product quality (long term):
- Extendibility
- Reusability
- Portability
- ...

The Software Engineering problem

Developing software systems that are
- On time and within budget
- Of high immediate quality
- Possibly large and complex
- Extendible
Lifecyle models

- Origin: Royce, 1970, Waterfall model
- Scope: describe the set of processes involved in the production of software systems, and their sequencing
- "Model" in two meanings of the term:
  - Idealized description of reality
  - Ideal to be followed

Models and standards

Capability Maturity Model (CMM)

- Characterization of maturity of the software development model of a company
- Five levels
- Popular with defense contractors, outsourcing companies
- Also: ISO 900x quality standards (International Standards Organization)

The anti-process movement

"eXtreme Programming" (XP), "Agile" methods

- Test-driven development
- Recommended practices, e.g. Pair Programming
- Short iteration cycles

"The revenge of the cubicles"

Arguments for the waterfall

(After B.W. Boehm: Software engineering economics)

- The activities are necessary
  - (But: merging of middle activities)
- The order is the right one.
End of lecture 20