Correctness

Hao Zhe Kok, Ramsey Towell, Binbin Wu

Correctness

- Definition: If the specifications of the systems are available, correctness assumes that it is possible to determine unambiguously whether a program meets the specifications.

- A program is functionally correct if it behaves according to its stated functional specifications. But may be missing other specifications such as performance and scalability.

- Basically if the software solves the problem if the problem is sufficiently defined.

- Must be unambiguous, can be assessed. Is a mathematical property that establishes the equivalence between the software and its specification.

Correctness

- Specifications are usually specified by people informally using natural language.

- This presents ambiguity because natural language is not consistent.

- Despite this ambiguity, specifications are useful to determine the approximate desirable goals of a given software system.

Correctness

- Correctness can be enhanced by using appropriate tools.

- Such as high-level languages - particularly those supporting extensive static analysis - testing using tools without running the program.

- Likewise, correctness can be improved by using standard proven algorithms or built-in libraries, instead of reinventing the wheel.

- Using proven methodologies and processes.