Evaluation of Computer Programs

- Is it correct?

- How easy is it to read and understand the code? Is it well documented?

- How easy is it to make changes to the program?

- How much memory is it needed to run the program?

- For how long will the program run?

- How general is the code? Will it solve problems over a large range of inputs without modification?

- Can the code be compiled and run on a variety of computers, or are modifications needed to run it on different computers?

Testing (Expose Errors)

- Establishing Correctness is quite difficult, but should be attempted at certain level.

- Exhaustive testing takes too long

- Designing test sets

  - Black Box Methods: Develop test data from function being computed.

  - White Box Method: Test data "covers the program".

Black Box Methods

- I/O Partition: Partition the I/O space into groups that behave similarly, and the behavior of data in different groups should be quite different (e.g. finding roots (2 roots, 1 root or imaginary roots)).

- Other methods: cause - effect graphing

White Box Methods

- Statement Coverage

- Decision Coverage

- Clause Coverage if((C1 && C2) || (C3 && C4))

- Execution path coverage

Debugging (Find and Correct errors)

- Use logical reasoning

- Don't correct errors by creating exceptions

- Make sure correction does not introduce errors where none existed.

- Test and debug incrementally.