

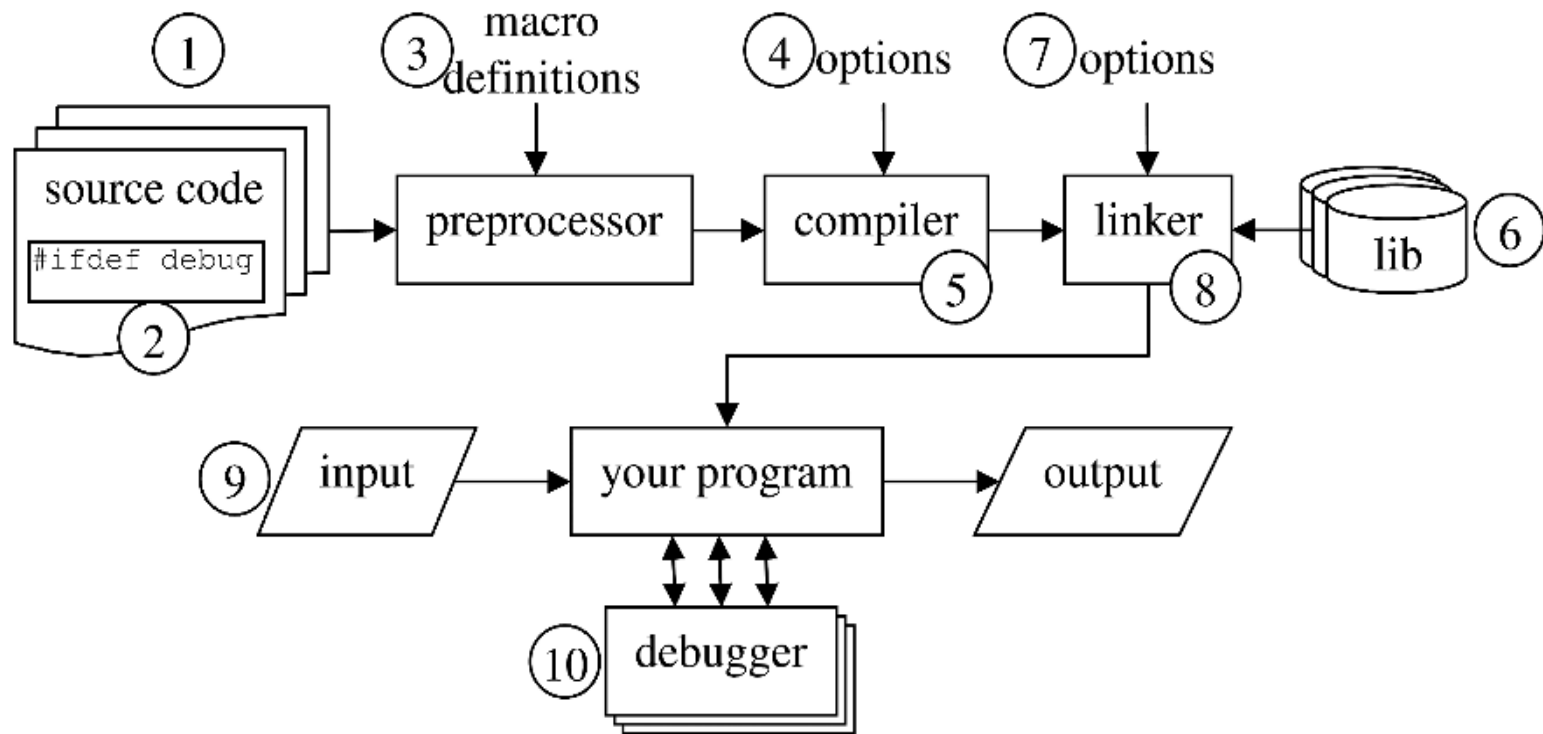
A Systematic Approach to Debugging

Presentation By : M.Imran
Memon

Debugging

- **Debugging** is a process of finding and reducing the number of bugs, or defects.
- **A Structured process Debugging** if you are unfamiliar with the system you are fixing bugs less chances for you to fix any error. You must know how the system works, what components it uses and what technologies are involved in development, what is the process of development.

Simplified build and test flow



Opportunities how to find Bugs

- Debuggable source code
- **Instrumentation** means writing additional code for observability and controllability, or to keep track, how often a function is called and by whom, of the software.
- These instrumentations are sometimes enabled by providing **macro** definitions.

Continue...

- **Compiler flags** can be used to generate code and information needed for source code debugging.
- **Static Checkers** Static checkers analyze the source code of the program without actually running the program.
- **Libraries** Selected libraries or third party libraries may also contains bugs that results in to abnormal execution of system.

Continue...

- **Linker options** Linking is the process of building an executable file from object files and object libraries.
- **Code Instrumentation tools** : Tools can be used to increase the accuracy upto the level of CPU cycles.
- **Test Cases** are designed to find the bugs from black spots.

13 Golden Rules of Debugging

- Understand the requirements
- Make it fail
- Simplify the test case
- Read the right error message
- Check the plug
- Separate facts from interpretation
- Divide and conquer

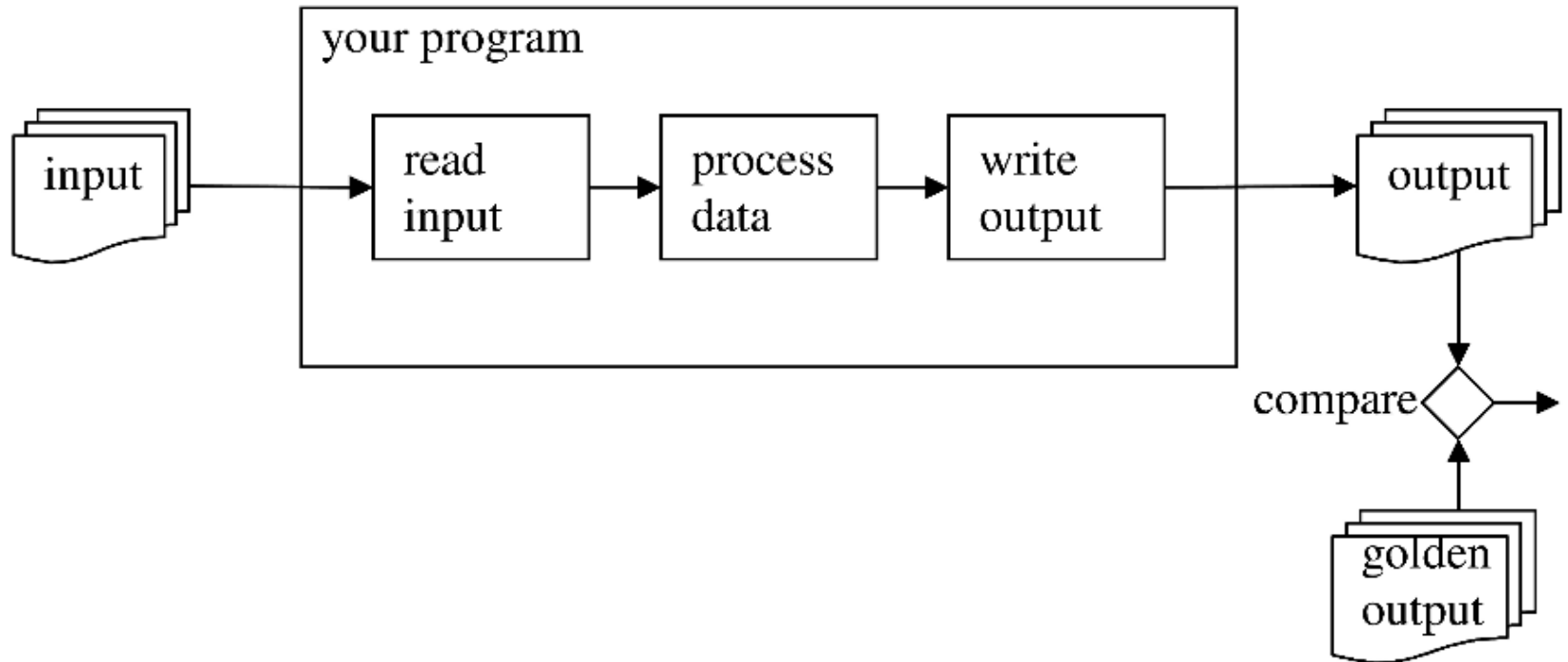
Continue...

- Match the tool to the bug
- One change at a time
- Keep an audit trail
- Get a fresh view
- If you didn't fix it, it ain't fixed
- Cover your bugfix with a regression test.

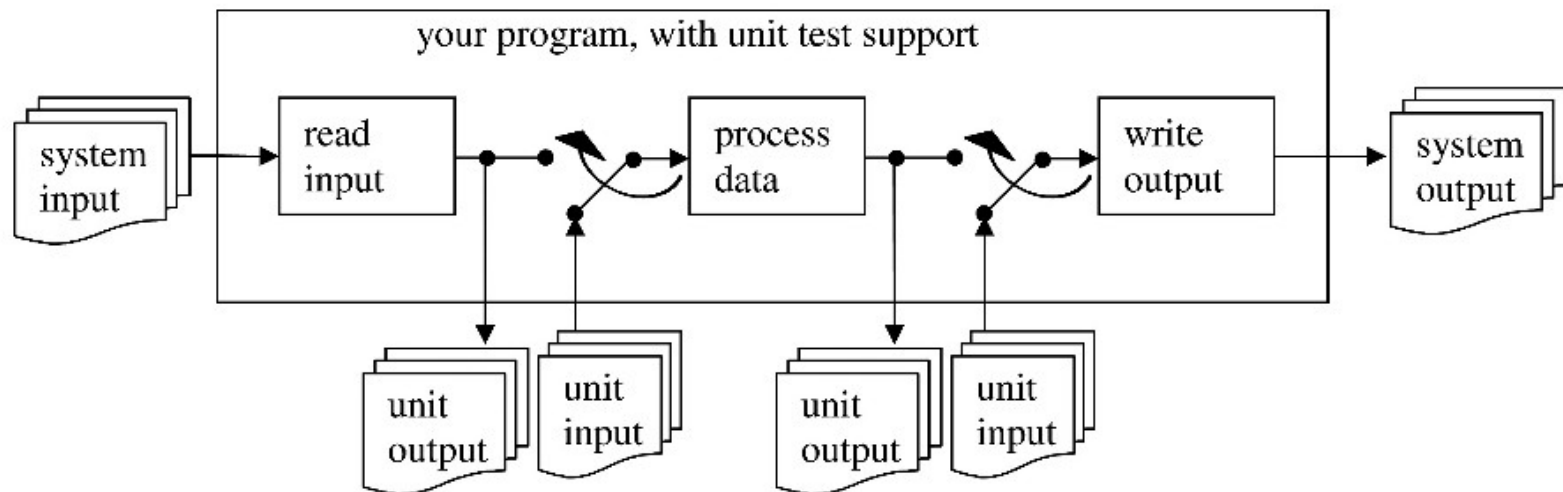
Build a Good Toolkit

- Your workshop
 - Running tests every day keeps the bugs at bay
3. Regression tests
 4. Unit tests and system tests.

System Testing



Unit Testing



Meet the Bug Family

- The Common bug
- Sporadic bug
- Heisenbugs
- 4. The Race-ist
- 5. The Memory Outlaw
- 6. The Idealist
- Bugs hiding behind bug
- Secret bug

Thank you