DARTMOUTH

Freezing Time and X-Ray Vision
Debugging Superpowers in Python

Simon Stone

Research Software Engineer for HPC and AI

Research Computing @ ITC, Dartmouth College

Introducing Research Software Engineering

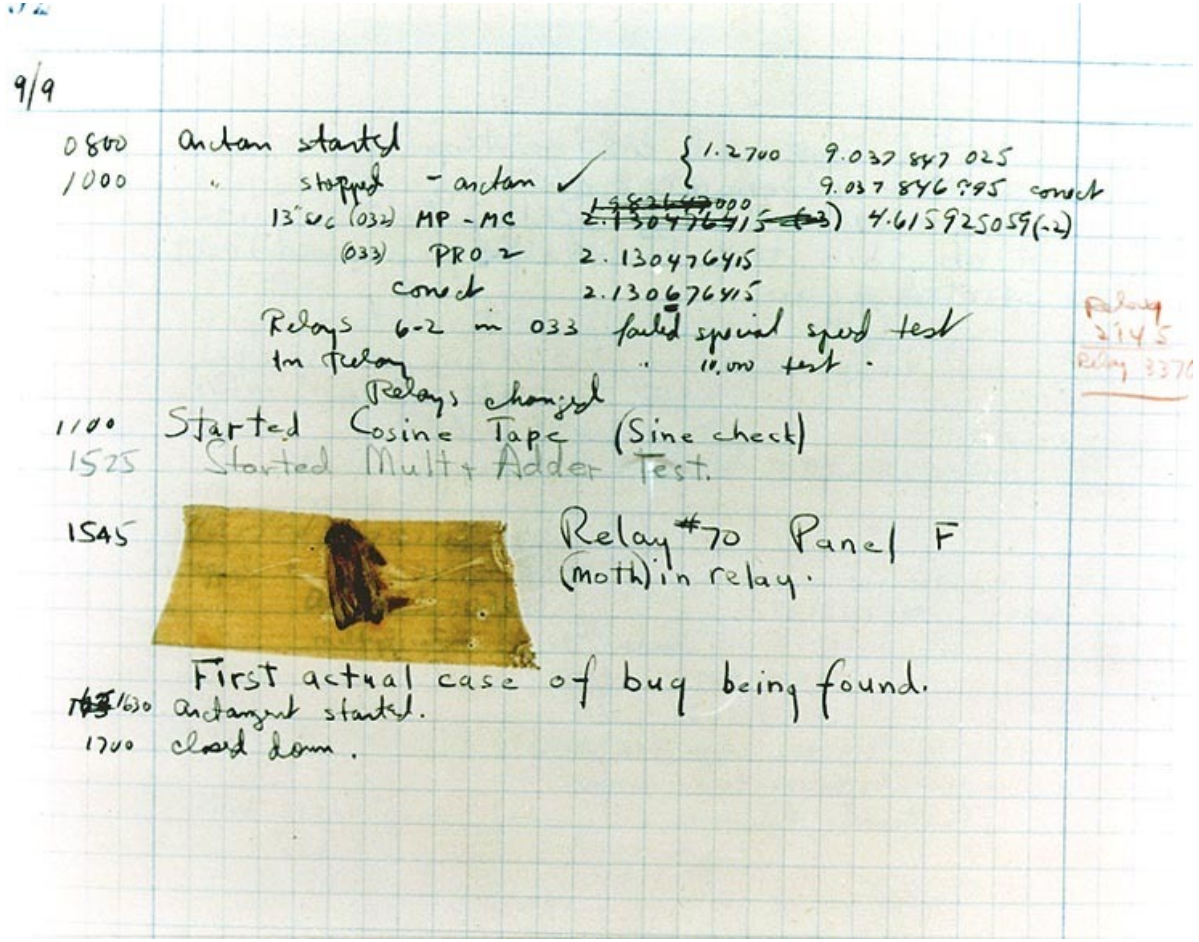
Collaborative expertise in software engineering, designed to bridge the gap between innovative ideas and impactful outcomes. Our services include:

- 🤝 **Grant Proposal Consulting** to ensure accurate resource estimations and project feasibility.
- 🚀 **Rapid Prototyping** to refine concepts and explore solutions.
- ⛑️ **Ongoing Application Support** and **Application Rehabilitation** for existing applications.
- 🌎 **Open-Source Releases** to share knowledge and contribute to the wider community.

Contact us today to discuss your project and discover how Research Software Engineering can be your trusted partner in innovation.

What is debugging?

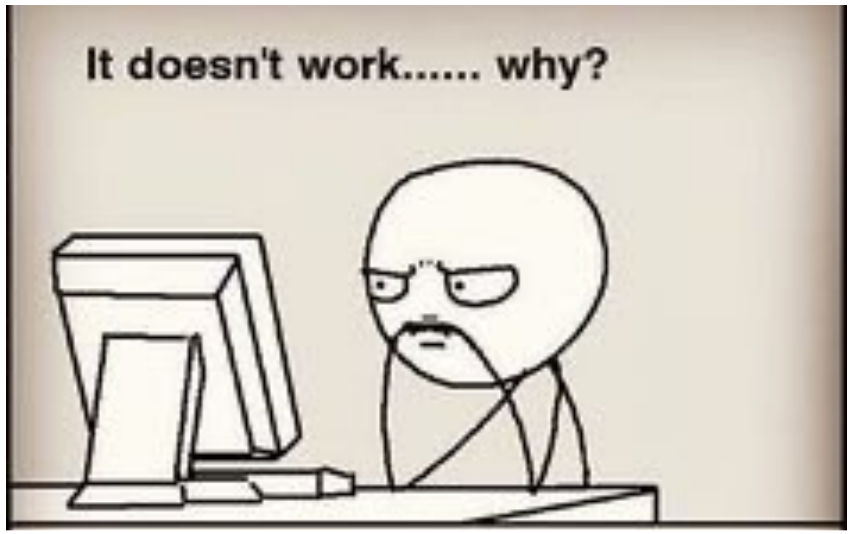
- 💥 Every programmer eventually hits an error in their program
  - 😵‍💫 Syntax errors, runtime errors, logic errors, …
- 🐛 “Debugging” is the process of systematically identifying and removing these bugs



Excerpt from Admiral Grace Hopper’s journal

Source: https://americanhistory.si.edu/collections/object/nmah_334663

How to debug?



- 🧠 Just read the code?
- 😵‍💫 `print` statements everywhere?
- 🧐 Write separate tests for everything?

- 🛠️ Virtually every programming language has a special tool for this purpose called the “debugger”!

Superpowers granted by today's workshop

- 💪 Core debugging skills
  - 🛑 Manage breakpoints: Freeze time!
  - 🩻 Inspect variables: X-Ray vision!
- 🎓 Advanced debugging techniques
  - 🧠 Use the debug console: Memory manipulation!
  - Analyze the call stack: Travel back in time!
- 👨‍💻 Professional development practice
  - 🦸 Integrate debugging into your regular coding workflow

The Debugger (in Python)

- 🪈 A program that sits between your code and the Python interpreter
- 👍 One such program comes with Python: pdb
  - 🐍 Lightweight and runs anywhere Python runs
  - 🤖 Command-line interface only and limited IDE integration
- VS Code comes with its own implementation called debugpy
  - Rich visual interface and easy to use
  - 🔒 Requires VS Code (or compatible editor)

The Debugger (in Python)

- 🔌 Debugger “attaches” to the Python process running your script
- 🛠️ Can pause execution, inspect and manipulate variables, …
- 🎮 A remote control for running your code!

Let’s get started!

Hands-on

Summary

- 🎉 The debugger is awesome!
- 💯 VS Code's implementation of the debugger is very helpful!
- 💡 The debugger takes the guesswork out of your error analysis

What's next?

- 🐸 Debug, debug, debug!
- ☝️ Debug someone else's code!

  Note: You may want to set "justmycode" to false in your launch.json
- 🚀 Learn to debug performance issues with profiling!
- 😎 Learn to debug parallel programs!

Thank you

dartgo.org/debugging-in-python