Debugging

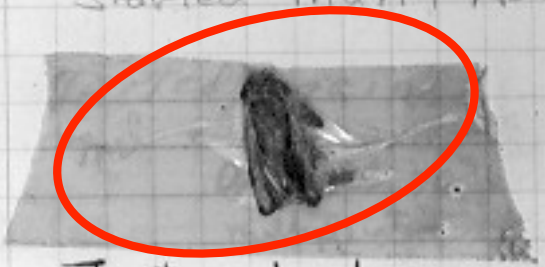
- The “history” of debugging
  - Often claimed that first bug was found by team at Harvard that was working on the Mark II Aiken Relay Calculator
  - A set of tests on a module had failed; when staff inspected the actually machinery (in this case vacuum tubes and relays), they discovered this:

92

9/9

0800 Arctan started { 1.2700 9.037 847 025
1000 " stopped - arctan ✓ 9.037 846 995 correct
13 wc (032) MP - MC ~~1.9826447000~~ 2.130476415 ~~(-3)~~ 4.615925059(-2)
(033) PRO 2 2.130476415
correct 2.130676415
Relays 6-2 in 033 failed special speed test
in relay " 10.000 test.
Relays changed

1100 Started Cosine Tape (Sine check)
1525 Started Multy Adder Test.

1545 
Relay #70 Panel F
(moth) in relay.

First actual case of bug being found.

~~1645~~ 1630 Arctangent started.
1700 closed down.

A real bug!

- However, the term bug dates back even earlier:
  - *Hawkin's New Catechism of Electricity, 1896*
    - "The term 'bug' is used to a limited extent to designate any fault or trouble in the connections or working of electrical apparatus."

Runtime bugs

- **Overt vs. covert:**
  - **Overt** has an obvious manifestation – code crashes or runs forever
  - **Covert** has no obvious manifestation – code returns a value, which may be incorrect but hard to determine
- **Persistent vs. intermittent:**
  - **Persistent** occurs every time code is run
  - **Intermittent** only occurs some times, even if run on same input

Categories of bugs

- Overt and persistent
  - Obvious to detect
  - Good programmers use **defensive programming** to try to ensure that if error is made, bug will fall into this category
- Overt and intermittent
  - More frustrating, can be harder to debug, but if conditions that prompt bug can be reproduced, can be handled
- Covert
  - Highly dangerous, as users may not realize answers are incorrect until code has been run for long period