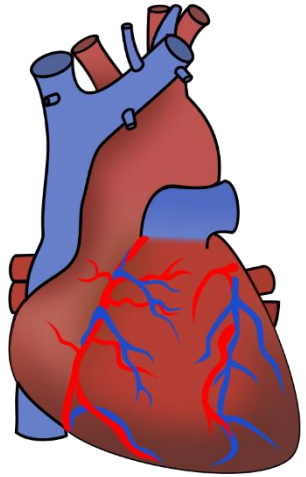
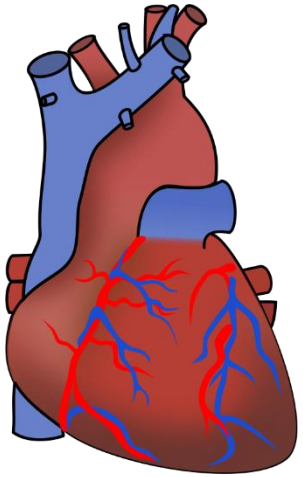
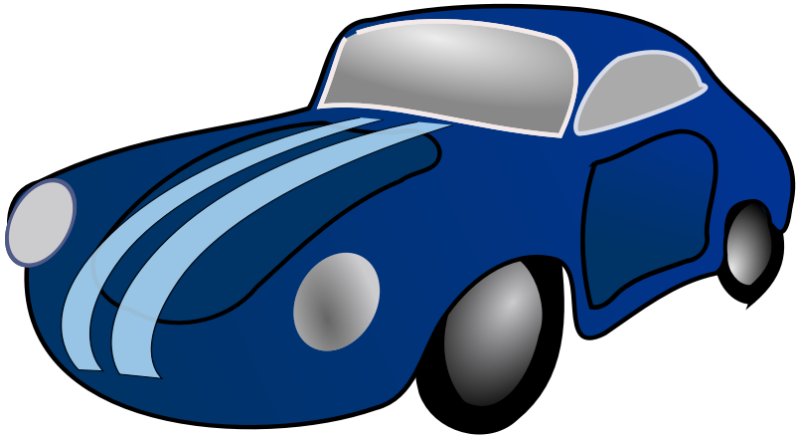
Testing and Confidence

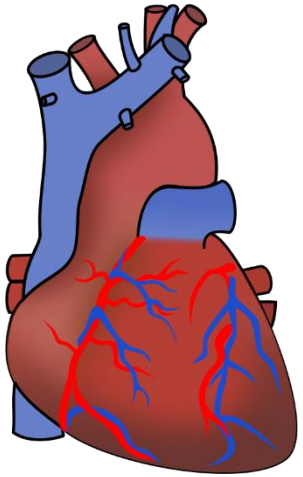
By the end of this video you will be able to…

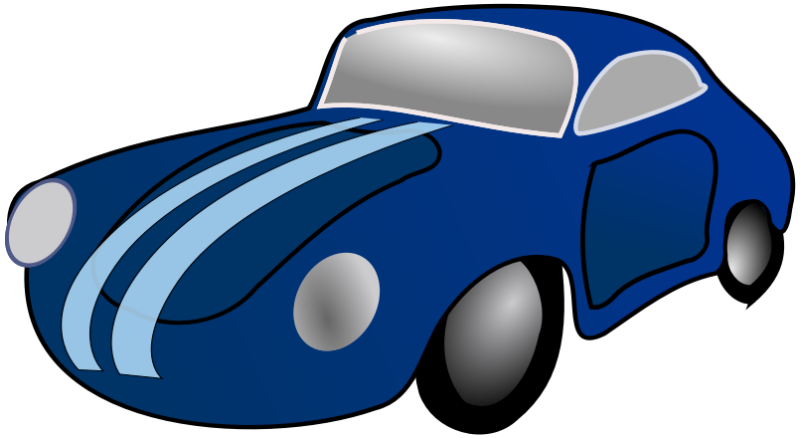
- Explain the value of having confidence in your code
- Identify ways to build confidence in code correctness

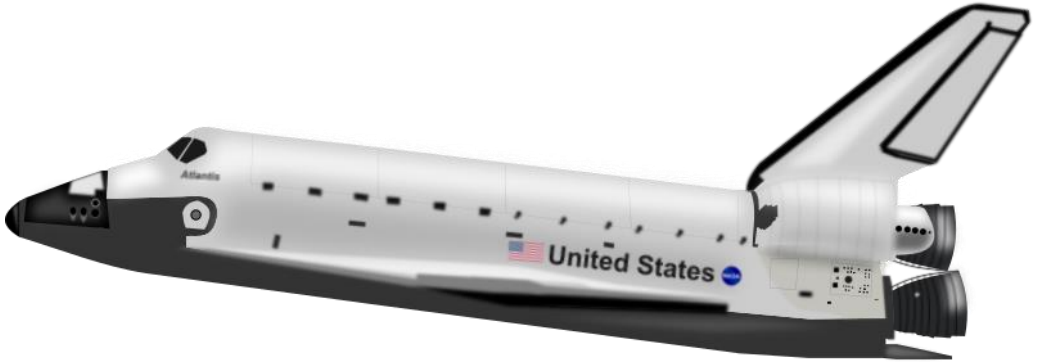


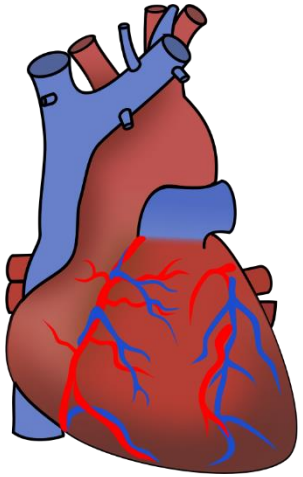


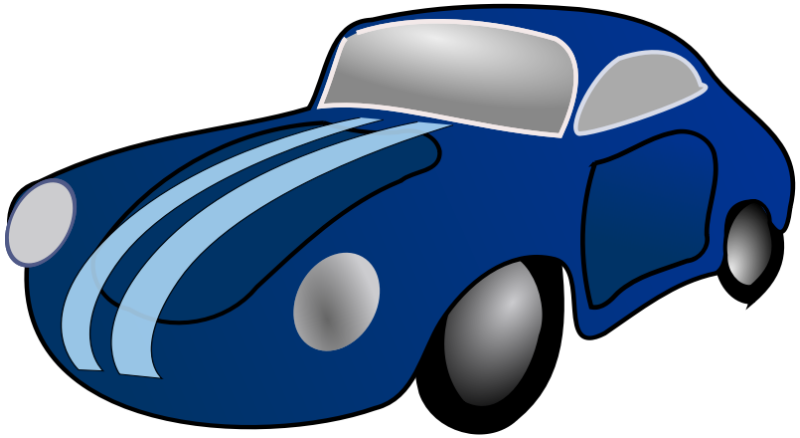


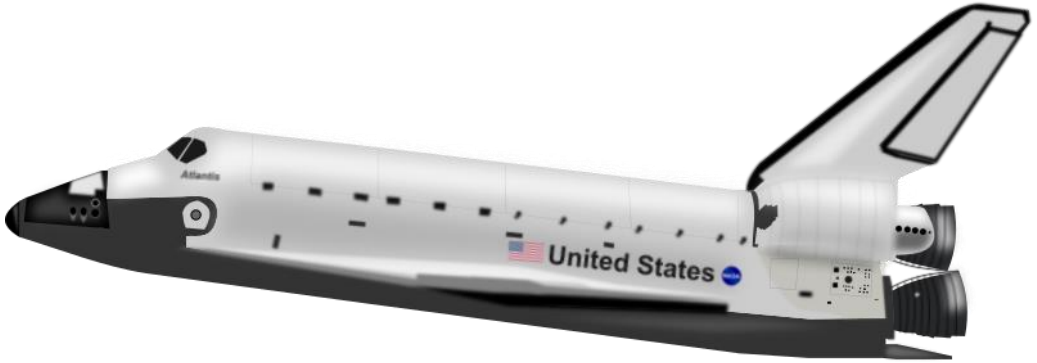




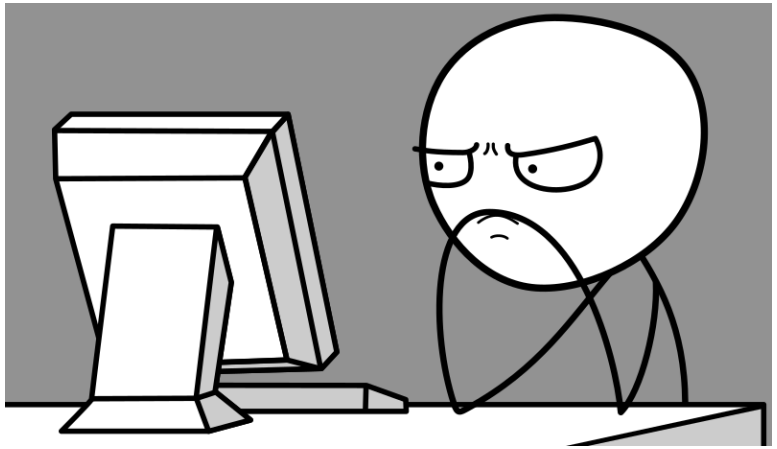

United States



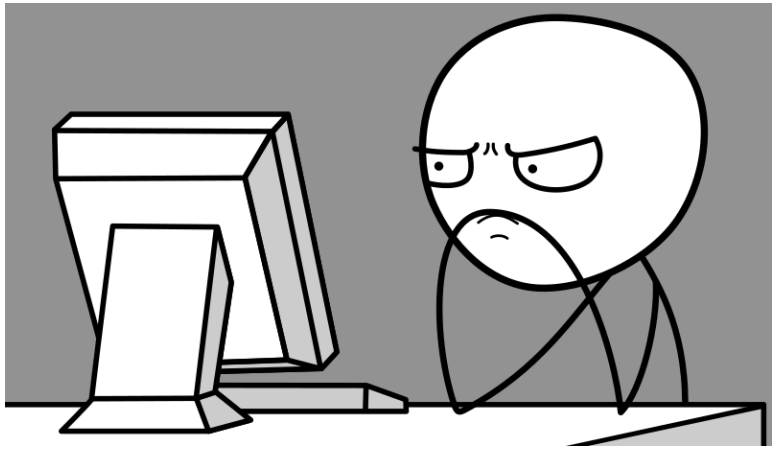




Confident in Correctness

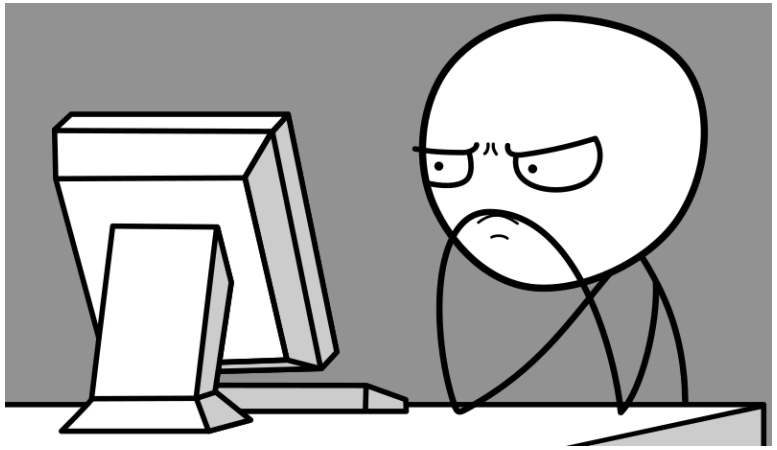


Users



Users

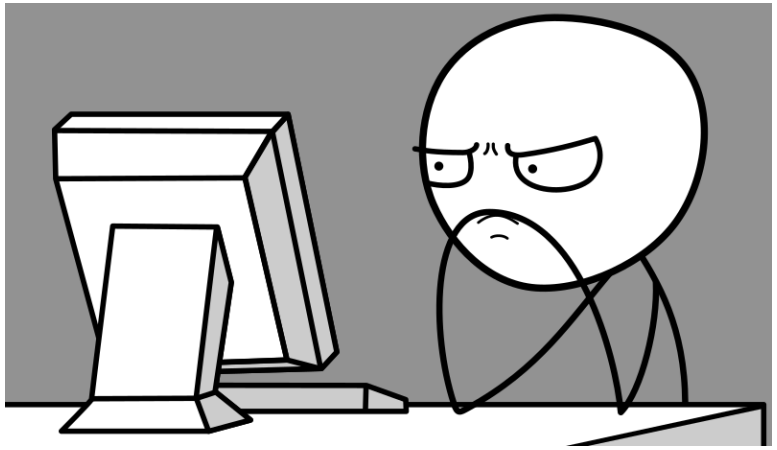
Hackers



Users

Hackers

Programmers



Users

Hackers

Programmers

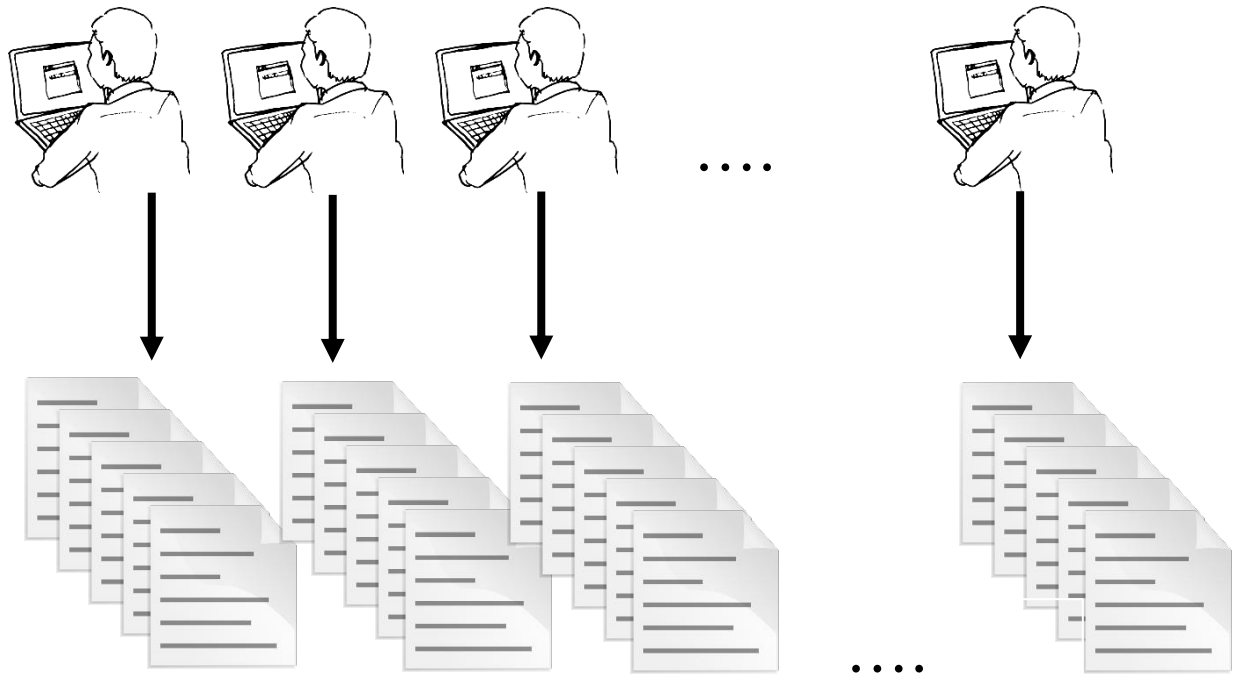
Yourself!

Programmers

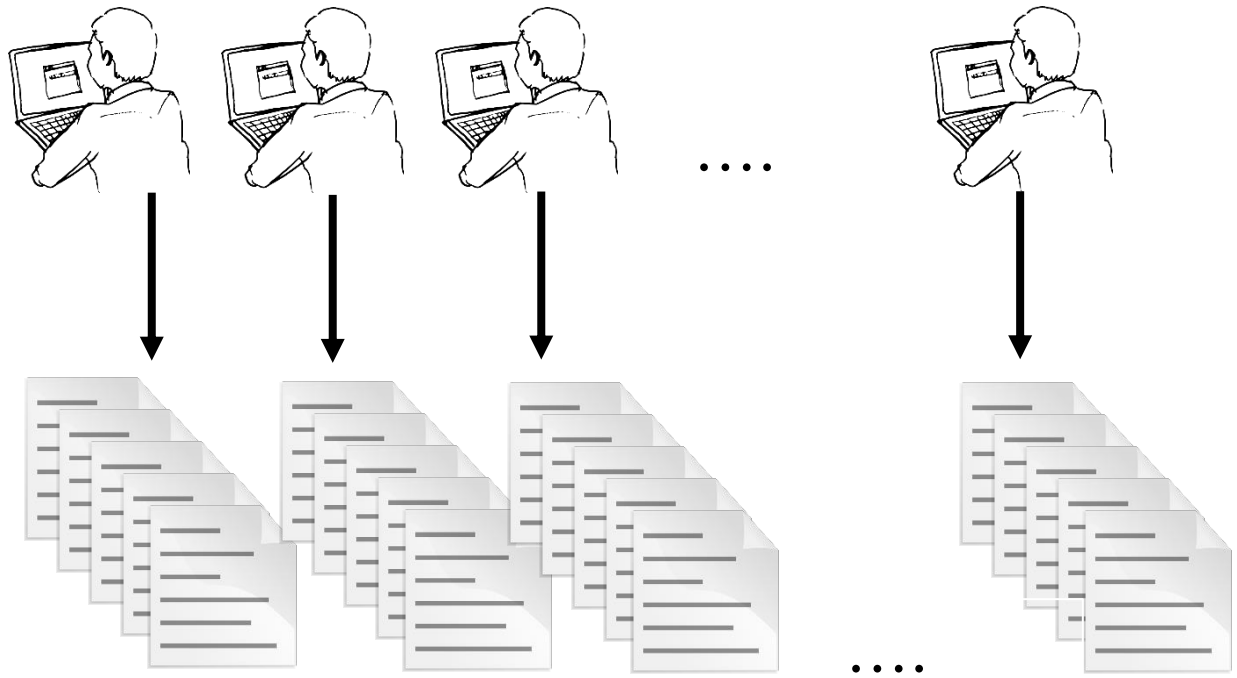
30
Programmers

30 Programmers

10 methods each



30 Programmers

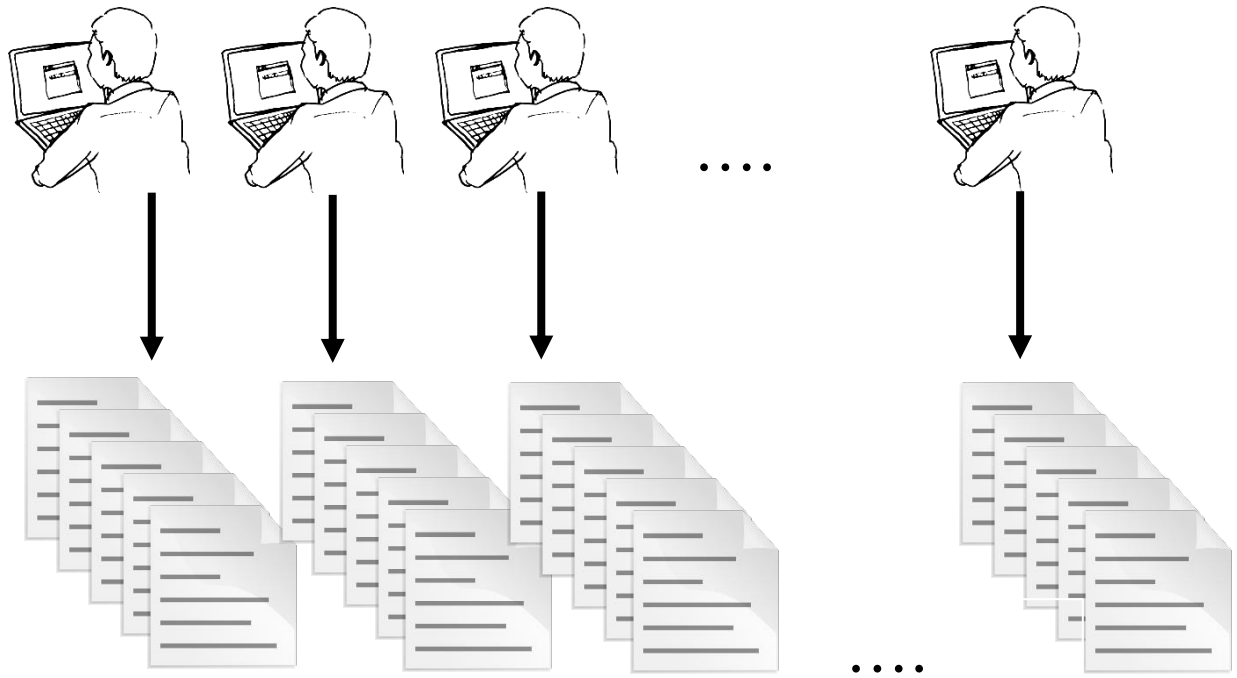


10 methods each

300 methods

99% chance correct

30 Programmers



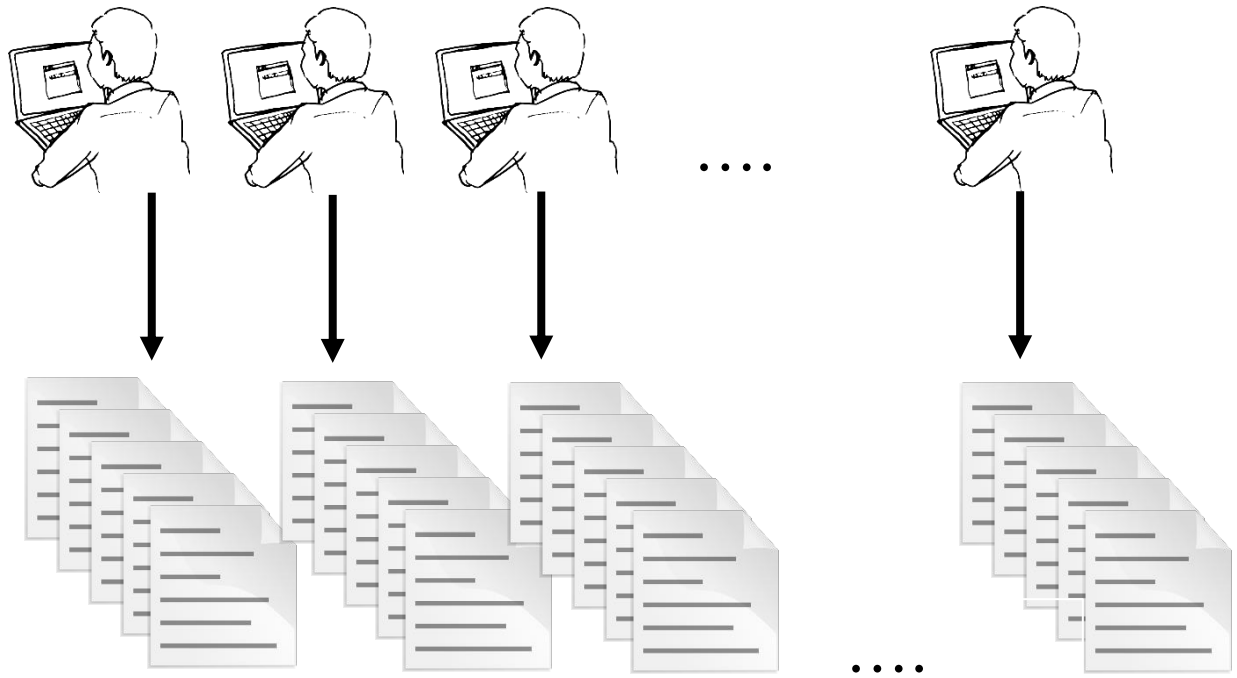
10 methods each

300 methods

99% chance correct

95% chance one is broken

30 Programmers



10 methods each

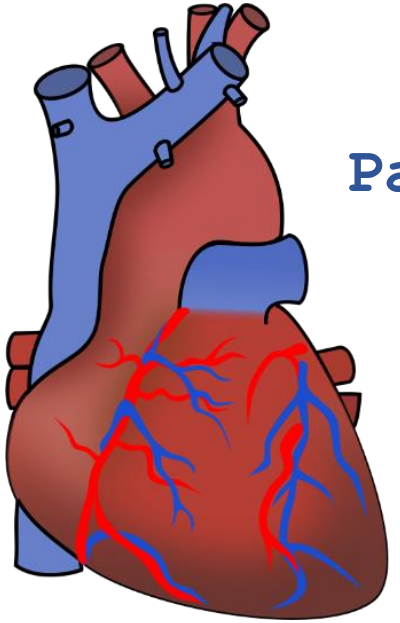
300 methods

99% chance correct

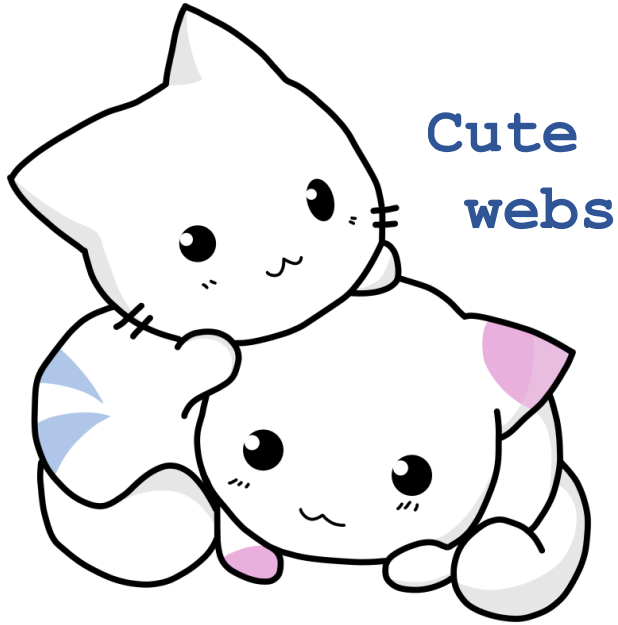
95% chance one is broken

Ignores interactions between methods

Another view? Risk Assessment...

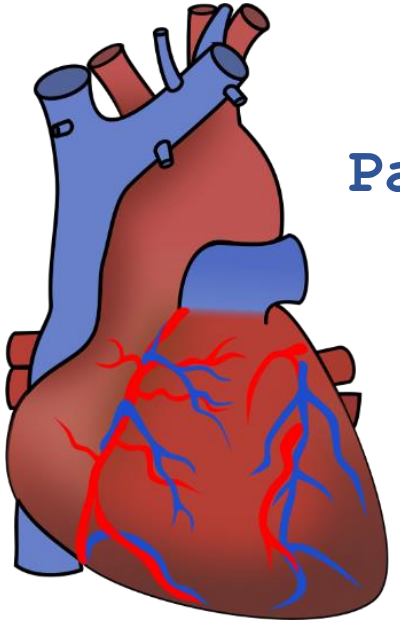


Pacemaker code

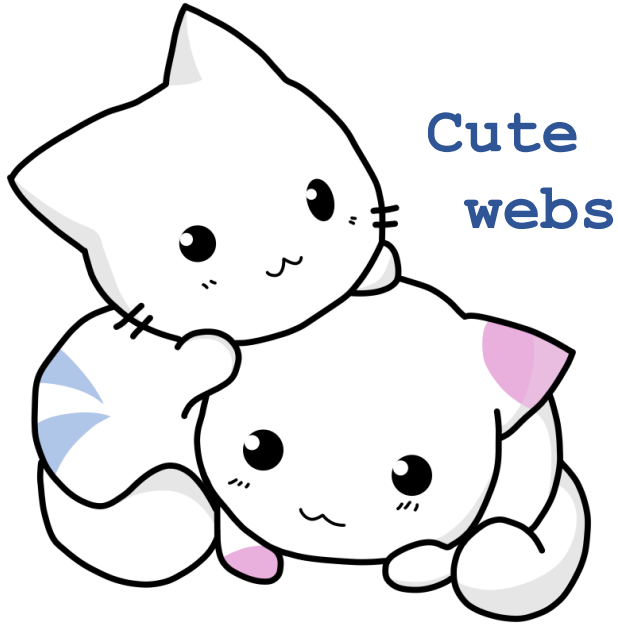


Cute Cats website

Another view? Risk Assessment...



Pacemaker code



Cute Cats website

Different degrees of confidence apply

Code State	Confidence
Written, hasn't compiled	Extremely low

Code State	Confidence
Written, hasn't compiled	Extremely low
Compiled, haven't run	**Extremely low**

Code State	Confidence
Written, hasn't compiled	Extremely low
Compiled, haven't run	Extremely low
Tested against basic input	**Low**

Code State	Confidence
Written, hasn't compiled	Extremely low
Compiled, haven't run	Extremely low
Tested against basic input	Low
Tested against corner cases	**Medium**

Code State	Confidence
Written, hasn't compiled	Extremely low
Compiled, haven't run	Extremely low
Tested against basic input	Low
Tested against corner cases	Medium
Tested against users	**Medium-High**

Code State	Confidence
Written, hasn't compiled	Extremely low
Compiled, haven't run	Extremely low
Tested against basic input	Low
Tested against corner cases	Medium
Tested against users	**Medium-High**

Wait, can't I just test against all inputs?

Testing against all inputs?

- If your method only depends on a boolean, sure.

Testing against all inputs?

- If your method only depends on a boolean, sure.
- What if the output depends on three ints?
  - An int has more than **four billion** possible values
  - Three ints have **79,228,162,514,264,337,593,543,950,336** possible combinations

Testing against all inputs?

- If your method only depends on a boolean, sure.
- What if the output depends on three ints?
  - An int has more than **four billion** possible values
  - Three ints have **79,228,162,514,264,337,593,543,950,336** possible combinations
- An array? A database? **Yikes!**

How can we increase confidence?

- Be critical of our algorithms/code
- Consider/test corner cases
- Attempt to formally reason about correctness
- Create automated test cases

How can we increase confidence?

- Be critical of our algorithms/code
- Consider/test corner cases
- Attempt to formally reason about correctness
- Create automated test cases