

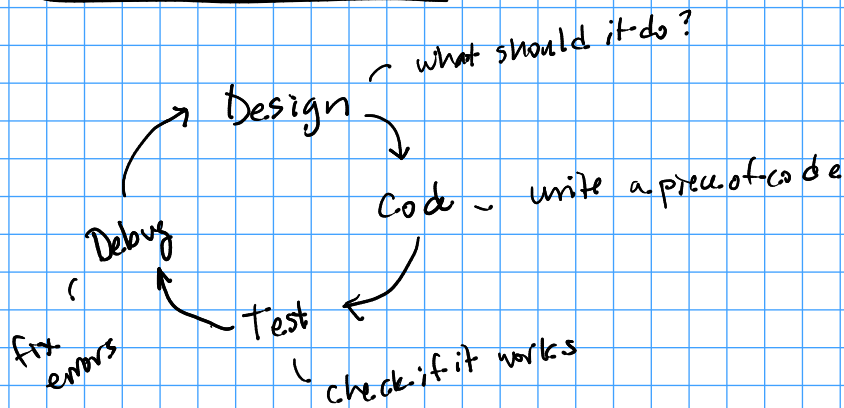
Lecture 9 - Debugging

* Programming Cycle

* Types of Bugs

* How to debug

* Programming Cycle



TIP write small pieces of code

* Types of Errors/Bugs

- syntax errors - "spelling", formatting
 - Easiest to find (interpreter catches them)

Q: example?

```
x=0  
while (x>5) {  
    x+=1  
}
```

- execution errors - code runs & crashes
 - ← something bad happens

eg. divide by 0

```
y <- 0  
x / y
```

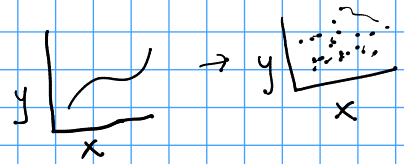
array[5] (only len = 4)

- medium difficulty

Q: example

- logical error - code runs to completion
But... it didn't do what you wanted.

- "computer is stupid"
- "evil genie"

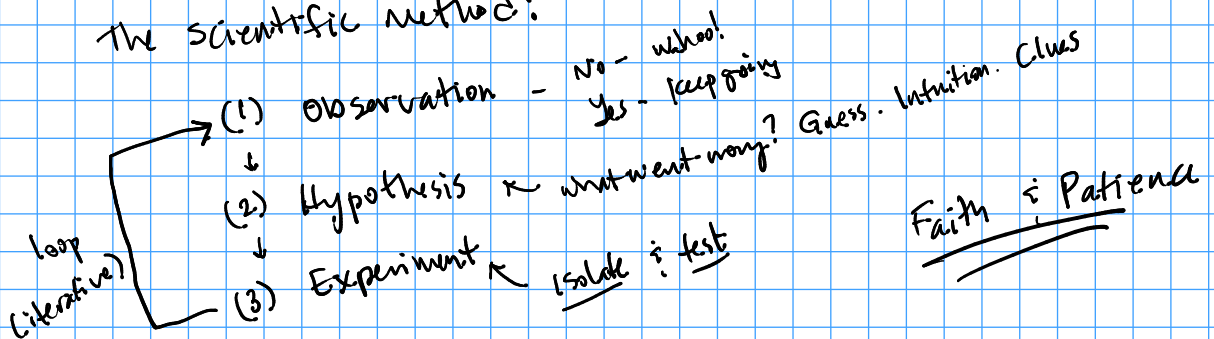


- hardest bugs to find and fix

Q: example

* How do I fix bugs?

The scientific method!



Tips - Good Hypotheses / Exp.

- Look at error message. (what line?) (How - array bad memory divide zero)
- Comment out lots of code - isolate
- Look at values of variables (prints, variable explorer)
- What are you assuming?
- Go to last line of code that worked.
- Write out on paper what you think it is doing

```
for i in range(5):
    print(i)
```

0

1

2

3

4

~~5~~

Activity

Observation
Hypothesis
Experiment

} HW - write this down

$A_{ij} \leftrightarrow A_{ji}$
↑ rows ← cols

$$A_{00} \rightarrow A_{00}$$

$$A_{01} \rightarrow A_{10}^+$$

$$A_{02} \rightarrow A_{20}^+$$