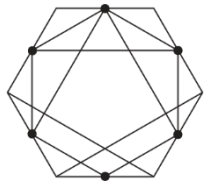
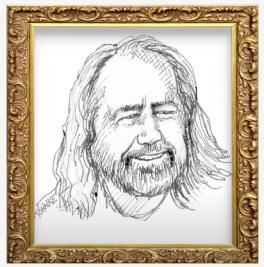
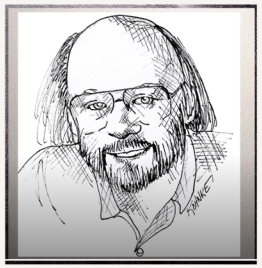


# Clean Code



Gary Booch

- Simple & Direct
- Reads as well-written prose



Bjarne Stroustrup

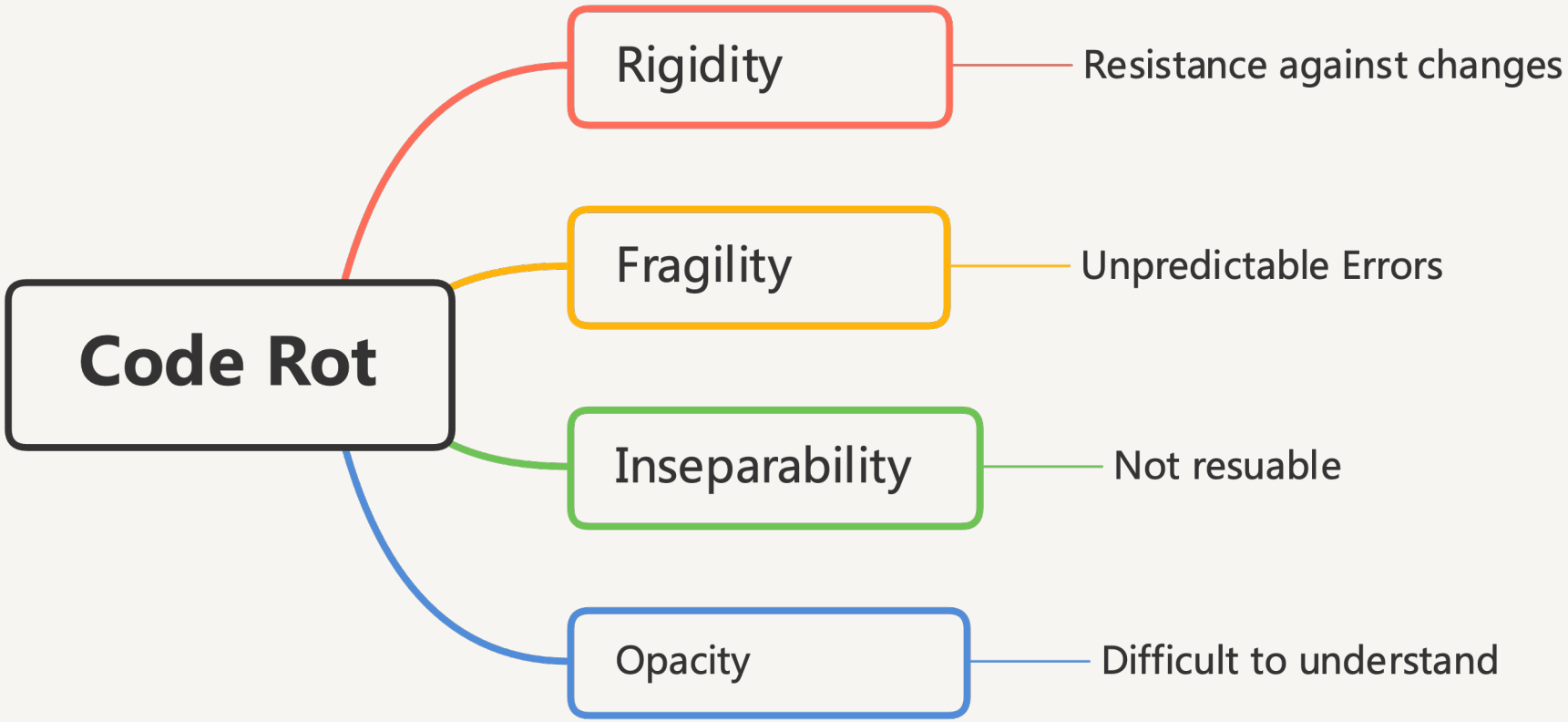
- Elegant & Efficient.
- Should do one thing

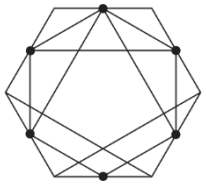
Ward Cunnigham

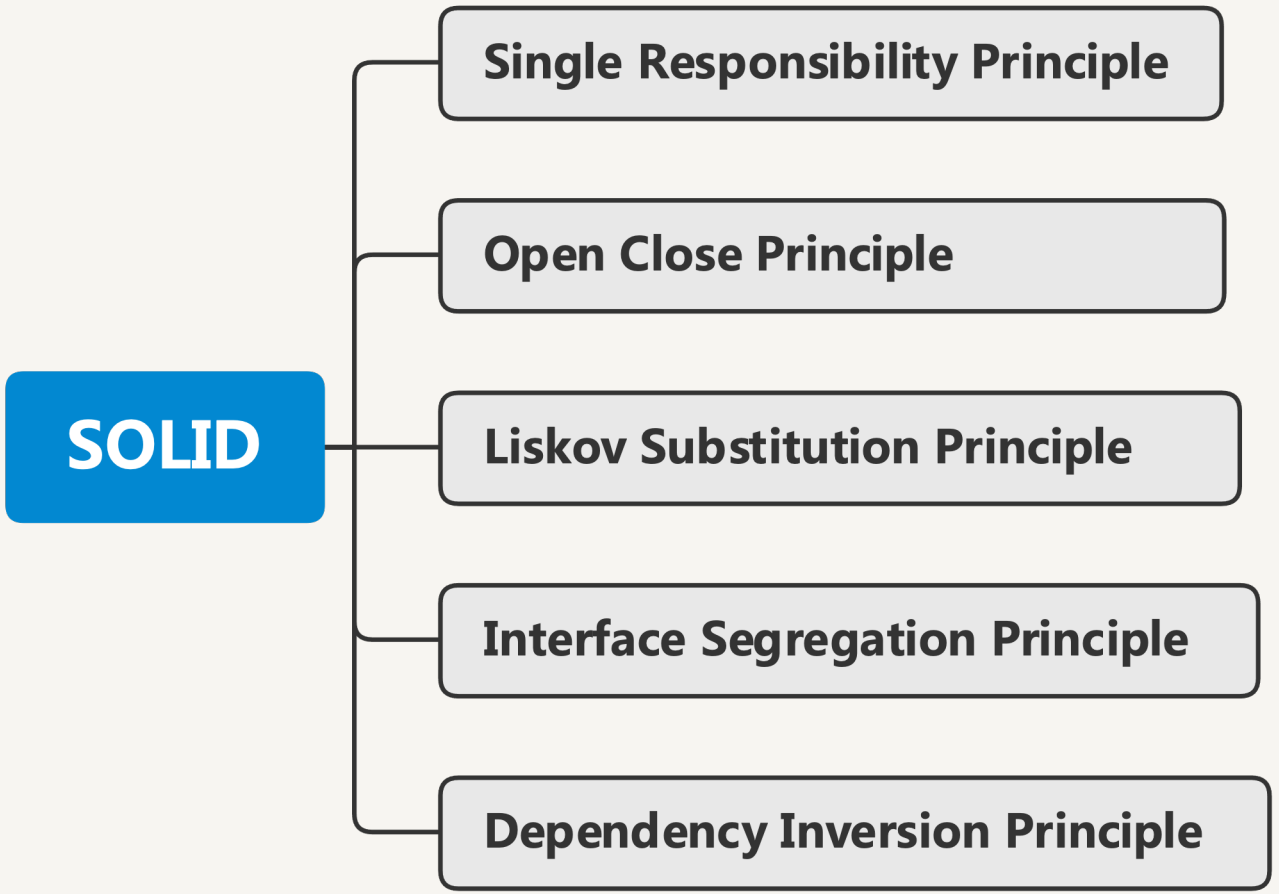
- Every Routine is pretty much what you expected

Michael Feathers

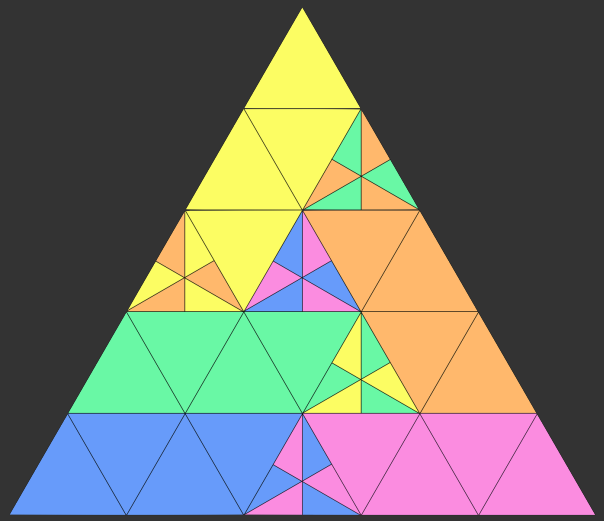
- Written by someone who cares



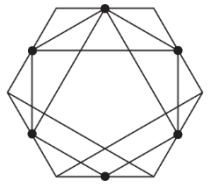




# INTERFACE SEGRETAION PRINCIPLE



- Do not depend on things
- You do not need!

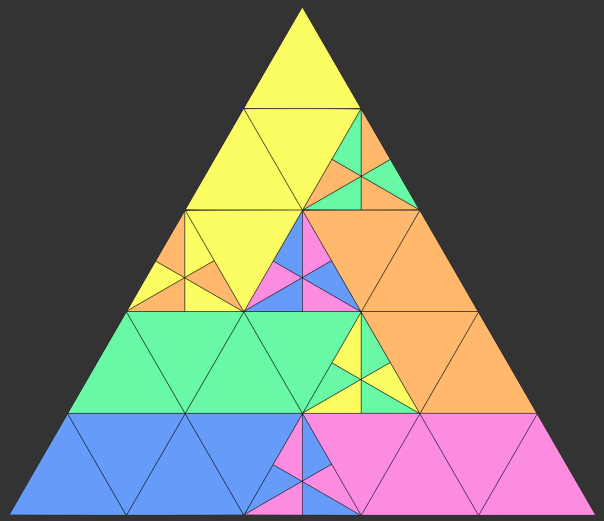


# INTERFACE SEGRETAION PRINCIPLE

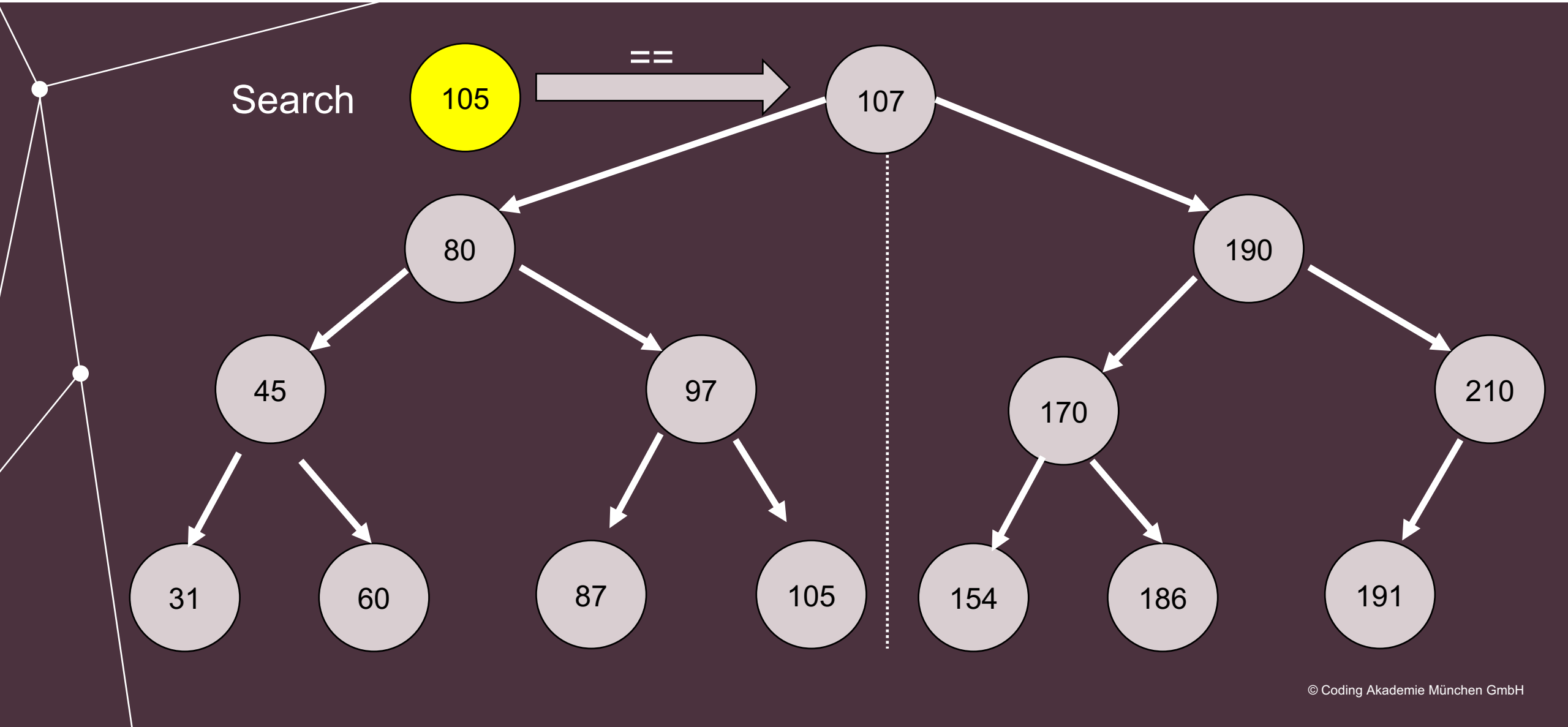
class C

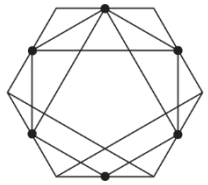
- No client of a class C
- Should depend on methods it doesn‘t use

# INTERFACE SEGRETAION PRINCIPLE



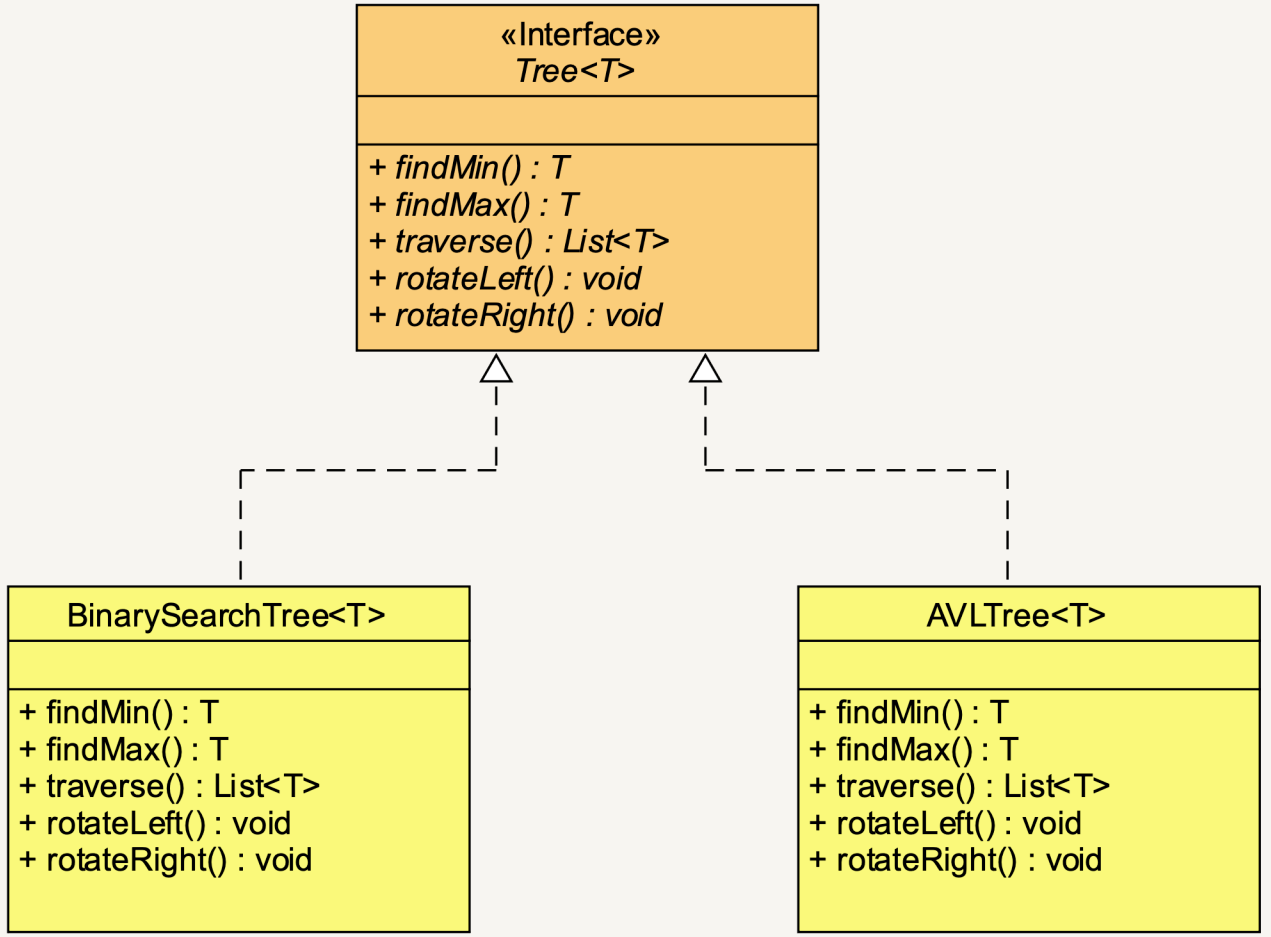
- If that is not the case
- Divide the interface of C into multiple independent interface
- Replace C in each client
- with the interfaces actually used by the client

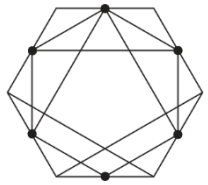




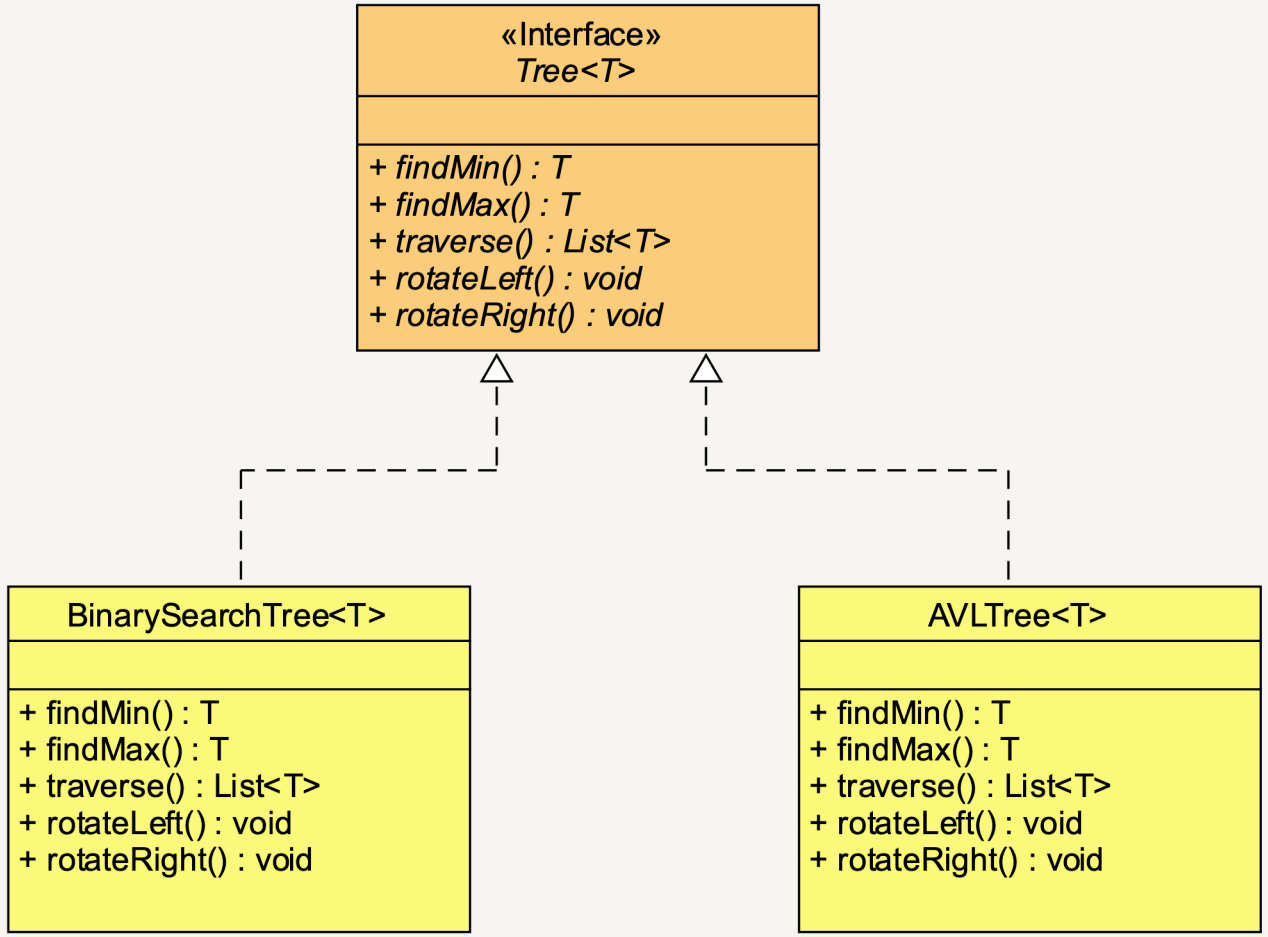
# WHY LOG(N)

- In each step we divide in half
- 32 => 16 => 8 => 4 =>2 => 1
- 32 = 2 ^ 5
- Run-Time log(32) = log(2^5) = 5

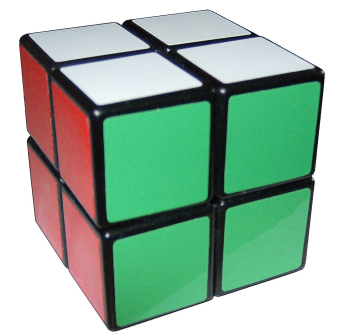


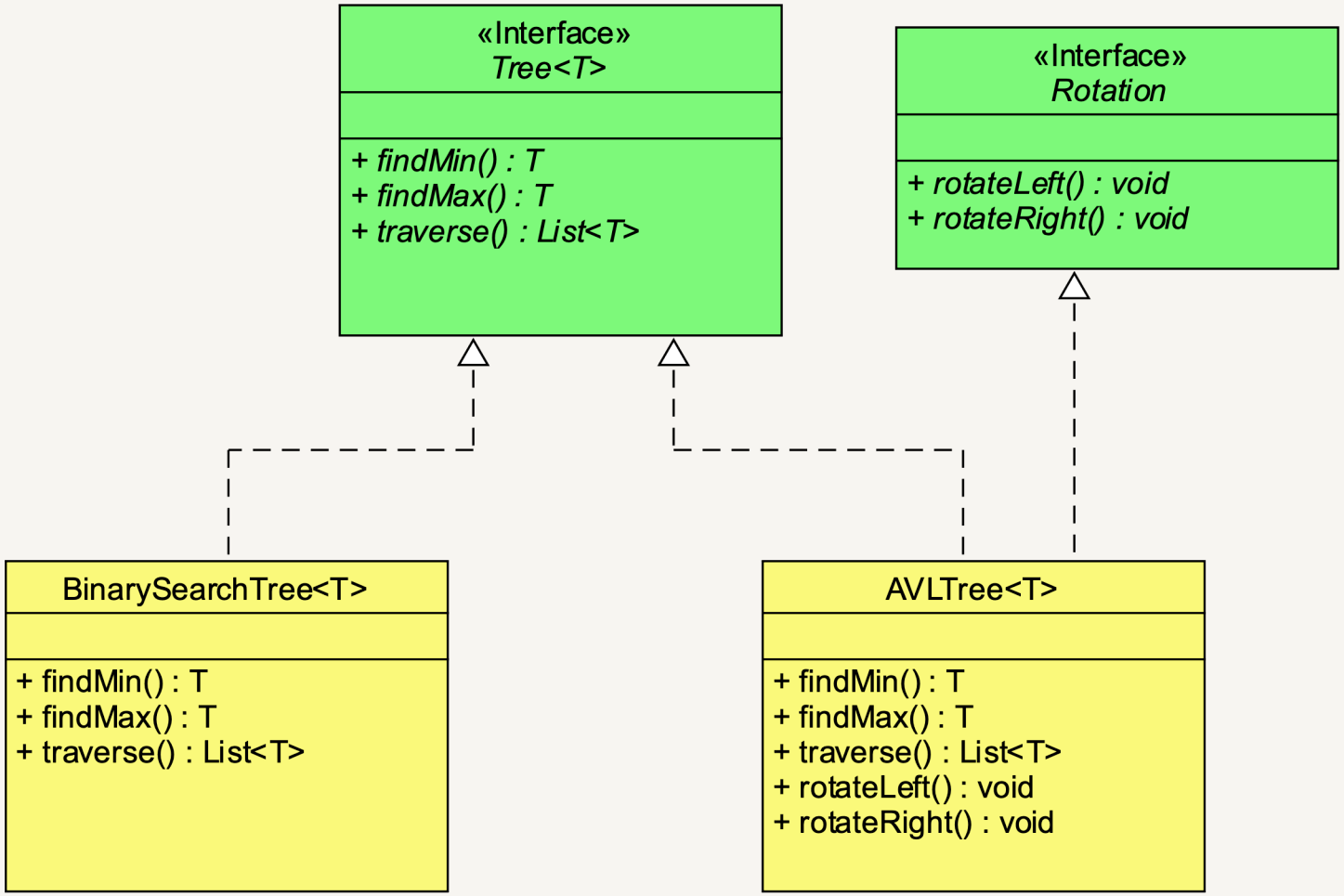


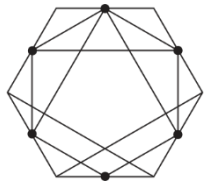
# ISP VIOLATION



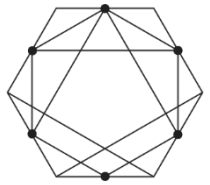
# SOLUTION



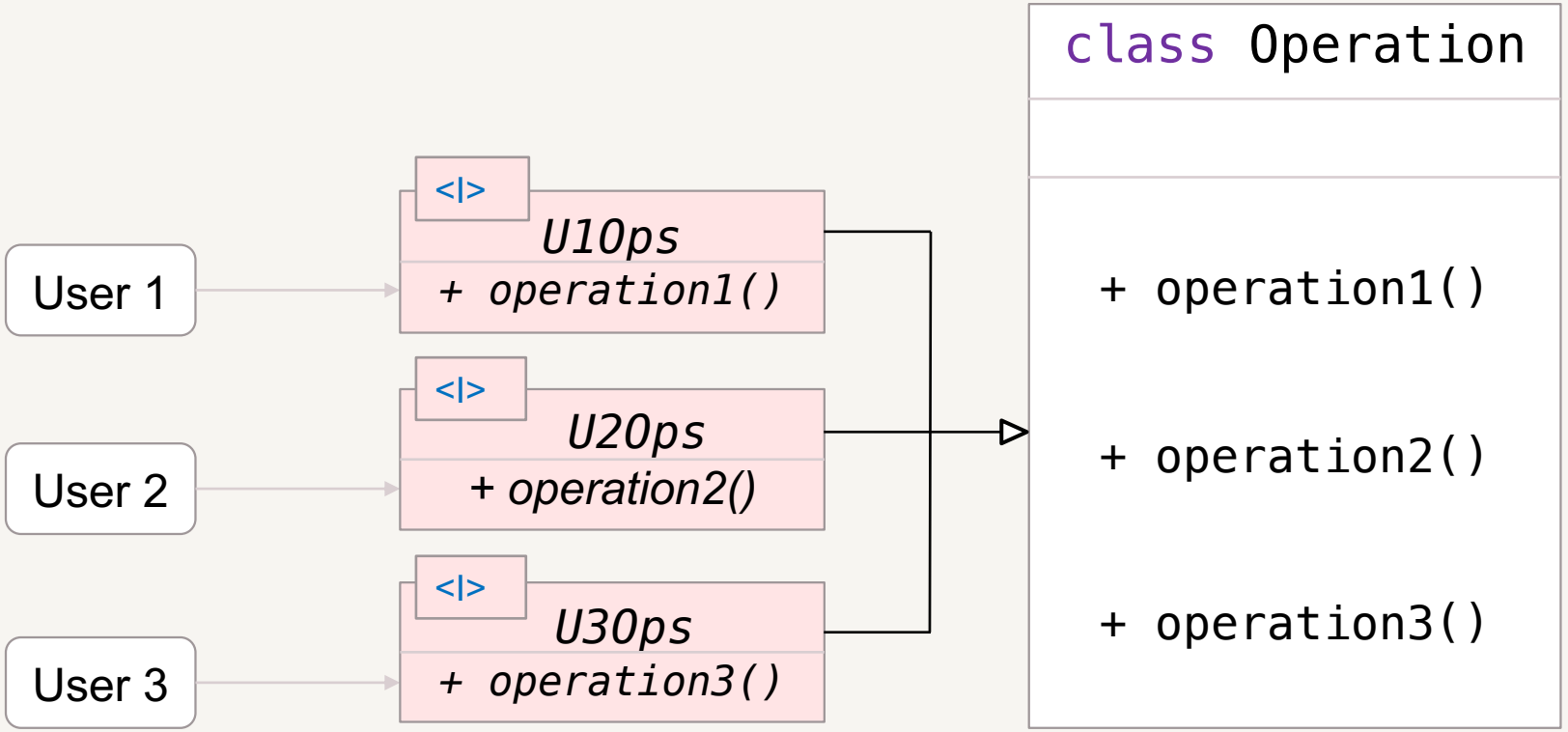




# ISP = DO NOT DEPENDS ON THINGS YOU DO NOT NEED!



# ISP = DO NOT DEPENDS ON THINGS YOU DO NOT NEED!



VIELEN DANK FÜR IHRE AUFMERKSAMKEIT.