Object Oriented Programming (OOP) Concepts

Abstraction

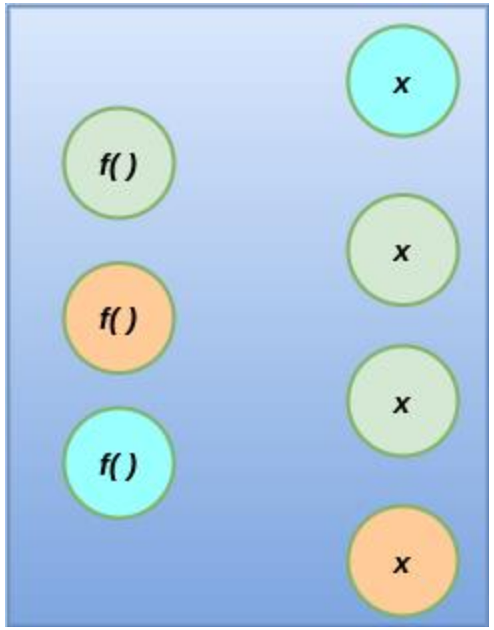
Encapsulation

Inheritance

Polymorphism

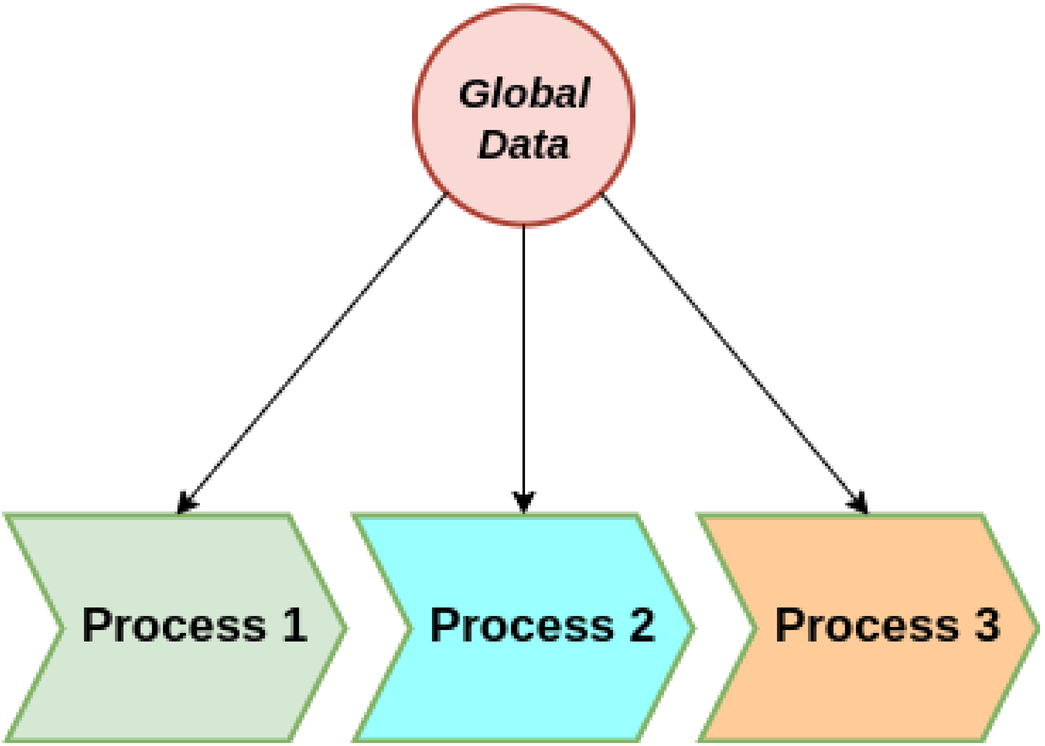
Procedural Programming

- Data is separate from methods
  - Data is usually global
  - Modified from one function to the next
- Functions are also separate
  - Can be complex
  - Usually requiring multiple parameters since they are distinct entities from data



Procedural Programming (2)

- How it usually works:

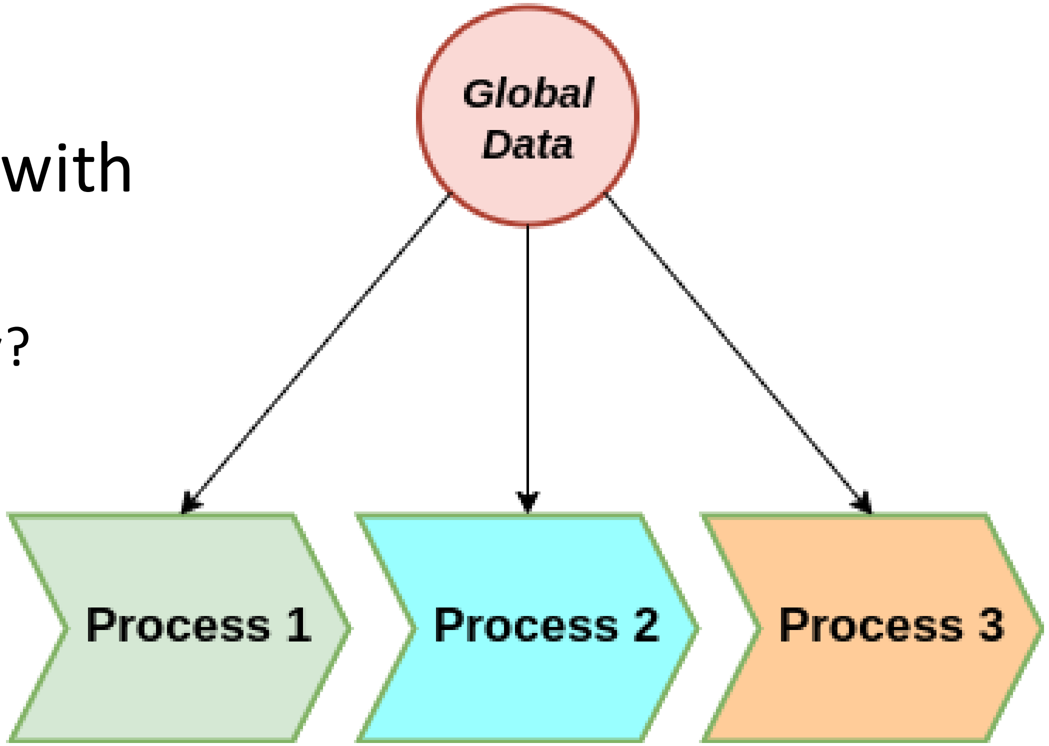


Procedural Programming (3)

- Advantages:
  - Easy to write (especially if short program)
  - Beginner-friendly

Procedural Programming (4)

- Disadvantages:
  - 1 Not always appropriate for large program
  - 2 Hard to represent a single “thing” (with behavior and states)
    - Can we represent a “person” as an entity?
    - Bank Account?
    - Car?



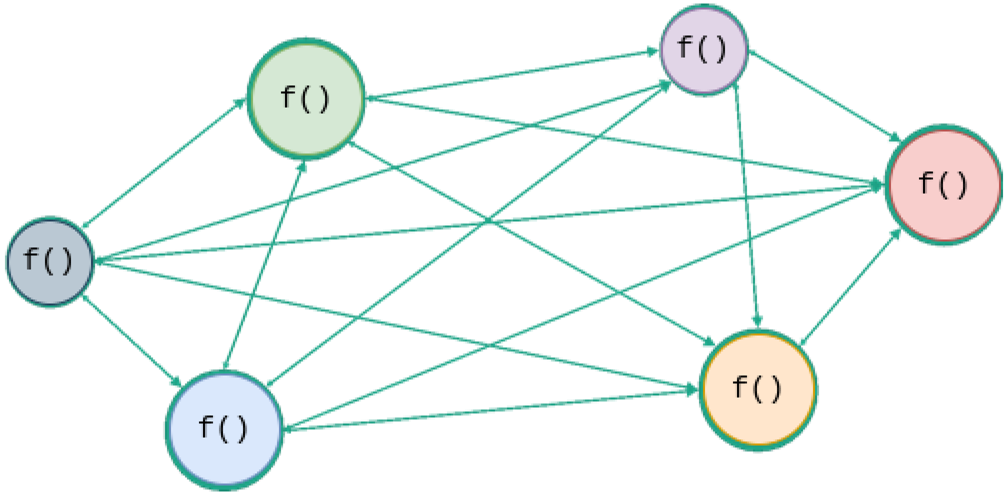
Procedural Programming (5)

- Disadvantages:
  - 3 Immutability problem
    - Ex: would it be good to modify the balance of a bank account from anywhere in the program?
    - What about your SSN?

Procedural Programming (6)

4

- Disadvantages:
  - Hard to maintain the entire program
  - Hard to assign portions of the code to different teams
  - One change can disrupt other modules

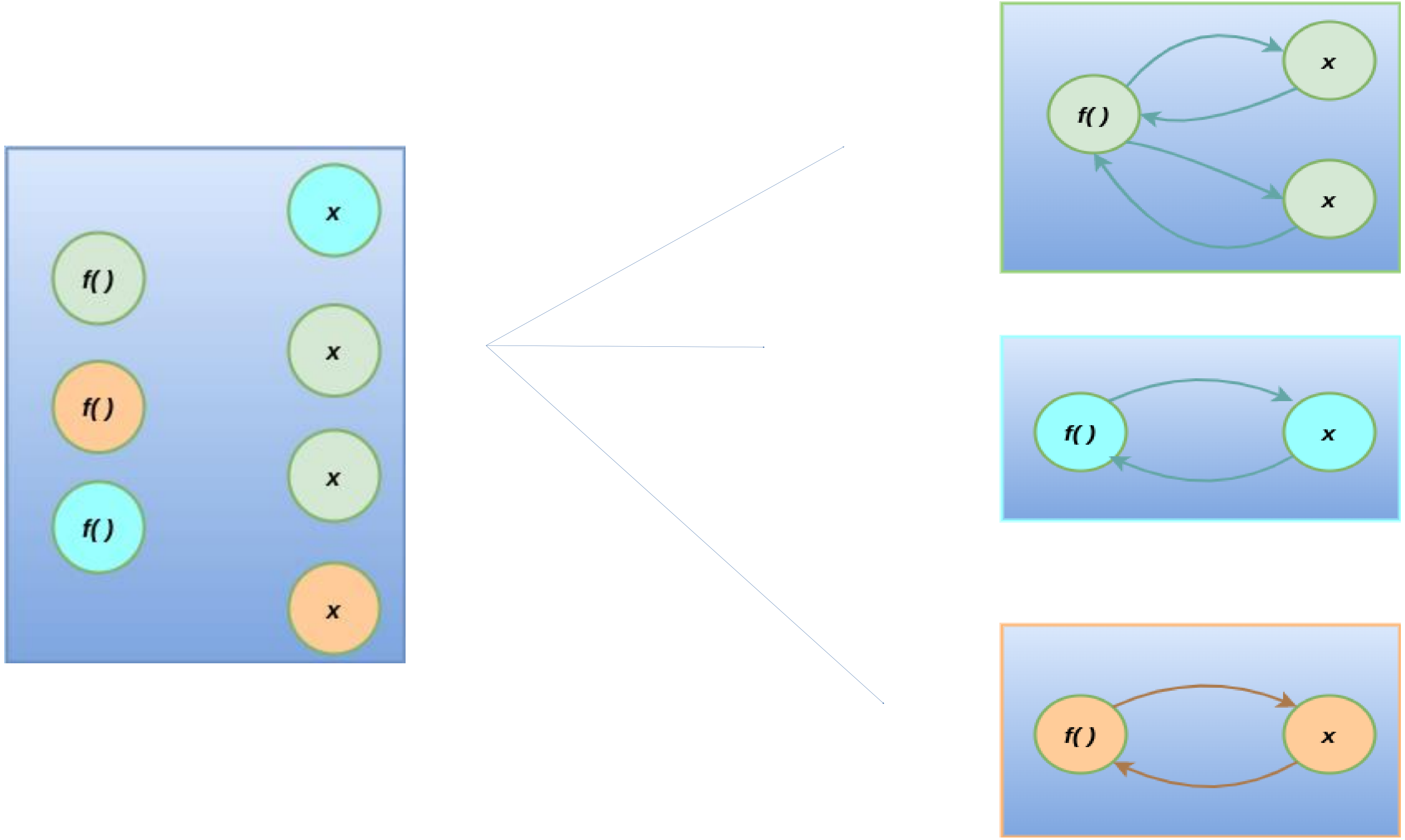


Avoid Spaghetti code!!

Object-Oriented Programming

- Centered around manipulating objects
  - Created from classes
  - Allows to bind data and methods
- Advantages:
  - 1 Can represent abstract things

Object-Oriented Programming



Object-Oriented Programming Pillars

Encapsulation

Abstraction

Inheritance

Polymorphism

- Increase reusability
- Reduce complexity

- Reduce Complexity
- Isolate the impact of change (pre/post conditions, information hiding)

- Eliminate code redundancy (child class can inherit parent class methods, behaviors, states) example: employee, students are also a person

- Code can take on multiple forms