Object Oriented C++

Michael Brodskiy

Professor: S. Shazli

March 27, 2023

- Object-oriented programming (OOP) is more natural to describe the interactions between "things" (*i.e.* objects)
- OOP provides better code reuse
  - Commonalities among objects described by a class
  - Commonalities among classes described by a base class (inheritance)
- Objects know what to do using their attributes: Each object responds differently to "What is your name?"
- OOP provides encapsulation: hide data that do not have to be visible to the other objects or protect data from unintentional, inconsistent changes
- Objects Definition
  - An object can be a "concrete and tangible" entity that can be separated with unique properties
  - An object can be abstract and does not have to be tangible
- Objects' Three Properties
  - The `Each` object is unique and can be identified (object's name) using name, serial number, relationship with another object, etc.
  - Each object has a set of attributes (data members), such as location, speed, size, address, phone number, on/off, etc.
  - Each object has unique behaviors (functions/methods), such as ring (phone), accelerate and move (car), take picture (camera), etc.
- Class Definition — Access Control
  - Information hiding — Encapsulation
    - To prevent the internal representation from direct access from outside the class
  - Access Specifier Keywords
    - `public`
      - May be accessible from anywhere within the program
    - `private`
      - May be accessed only by the member functions, and friends of this class, not open for non-member functions
    - `protected`
      - Acts as public for derived classes (child)
      - Behaves as private for the rest of the program

- Difference between classes and structs in C++
    - The default access specifier is private in classes
    - The default access specifier is public in structs