A workshop to share ideas, specifically “killer examples”, for Fundamental Object-Oriented (OO) principles and Design Patterns primarily in an objects-first CS1-CS2 sequence.

**Pedagogical Process**

1. Use it
2. Conceptualize it
3. Build it
4. Password web tutorial
5. Pedagogical progression
6. Evaluate software

**Caspersen’s Curve**

- Time for another killer example
- Time back in dorm...
- Time for motivation
- Time for reinforcement
- Time for understanding

**Problem Solving Process**

- Use an instance of Iterator
- General concept of an Iterator
- Define own Iterator
- Separate Iterator to access private parts of a collection

**Motivation for Teaching Design Patterns**

- Systematic way to solve complex problems
- Solutions that scale up
- Design patterns support building of correct, robust, flexible & extensible software in an efficient manner (time & $).

**Underlying Principles**

- Abstraction
- Invariant/variant decoupling (commonality/variability analysis)
- Parameterization
- Extreme encapsulation

**Wong’s Mapping**

- Underlying principles supporting our goals
  - Strategy
  - Adapter
  - Decorator
  - MFC

- Design patterns with problems (right or wrong)
- Decorator solutions
- Strategy, Adapter, Decorator