

# Designing Objects with Responsibilities

Curt Clifton

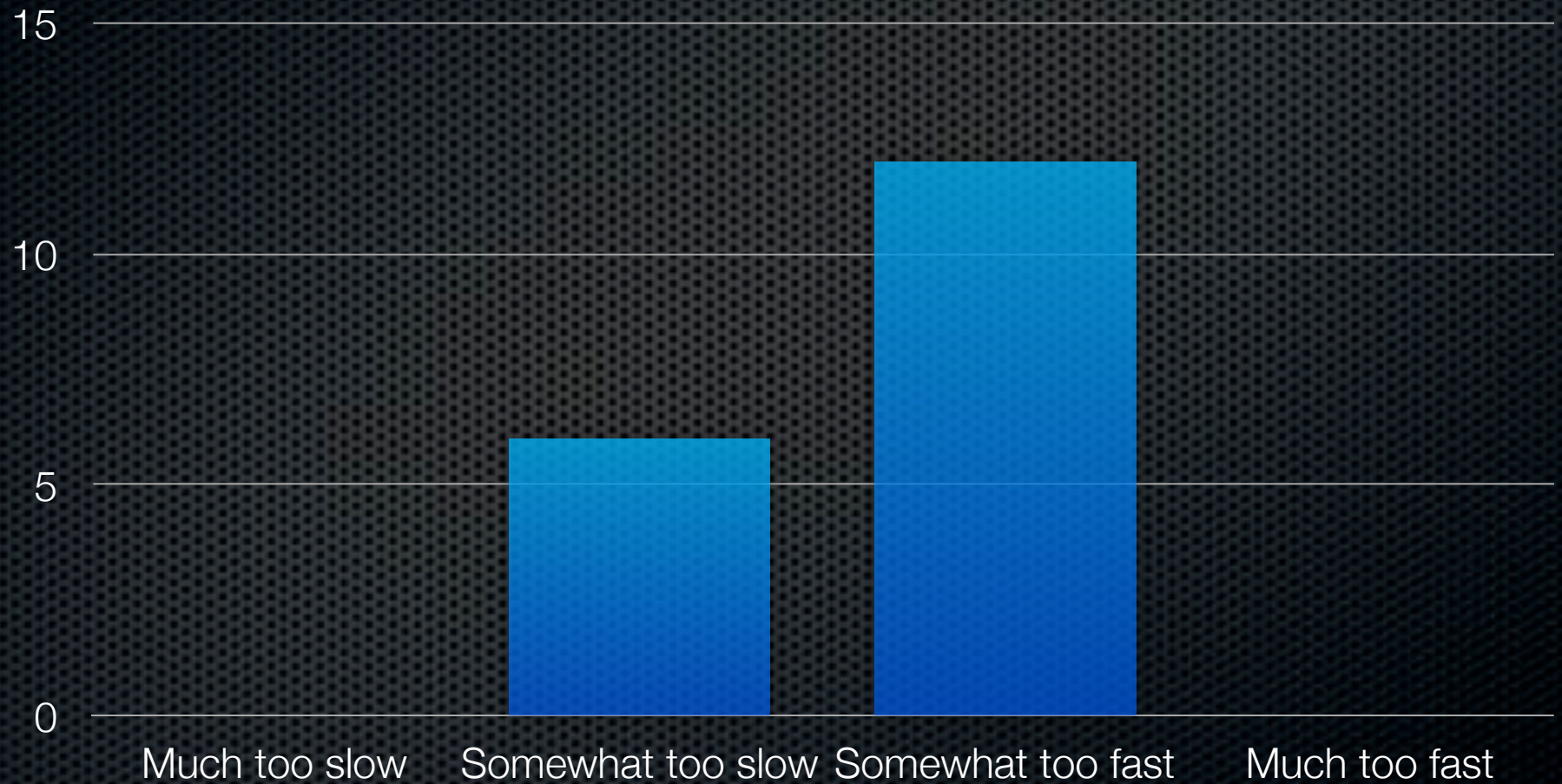
Rose-Hulman Institute of Technology

# Our Story Thus Far

- ✦ OO Analysis
  - ✦ Domain modeling
  - ✦ Identifying system operations
  - ✦ Clarifying critical operations
- ✦ Logical architectures, layers

# Pre-break Course Evaluation Results

# Lecture Pace



# Lecture +/-∂

+ In-class examples (9)

+ Slides (7)

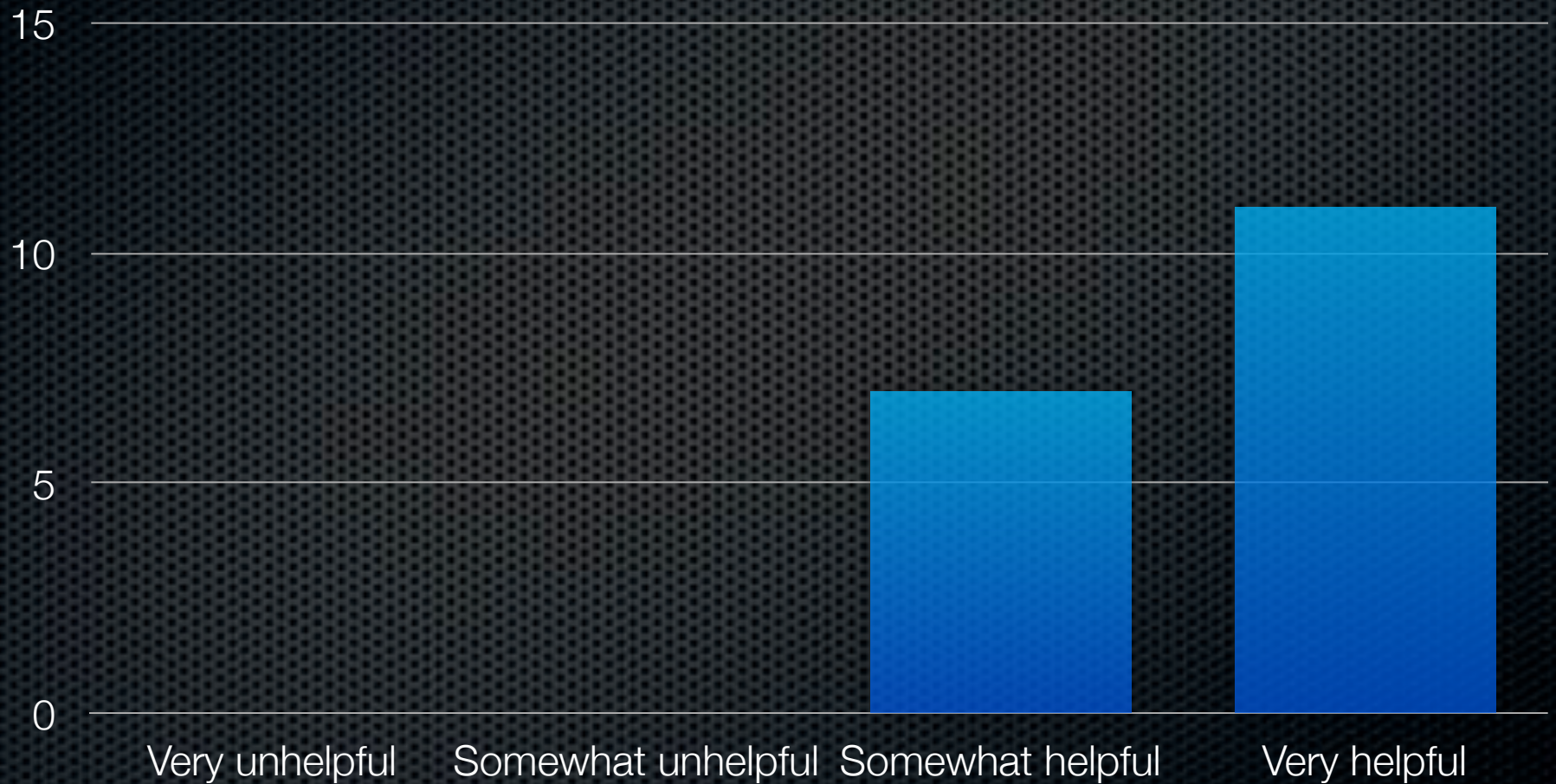
+ Diagrams on quizzes (4)

+ Style/enthusiasm (2)

∂ A bit less excited (2)

∂ Wrangle Raptors (2)

# Daily Quizzes Helpfulness



# Daily Quizzes +/∂

+ In-class examples (6)

+ Reference (6)

+ Help to pay attention (3)

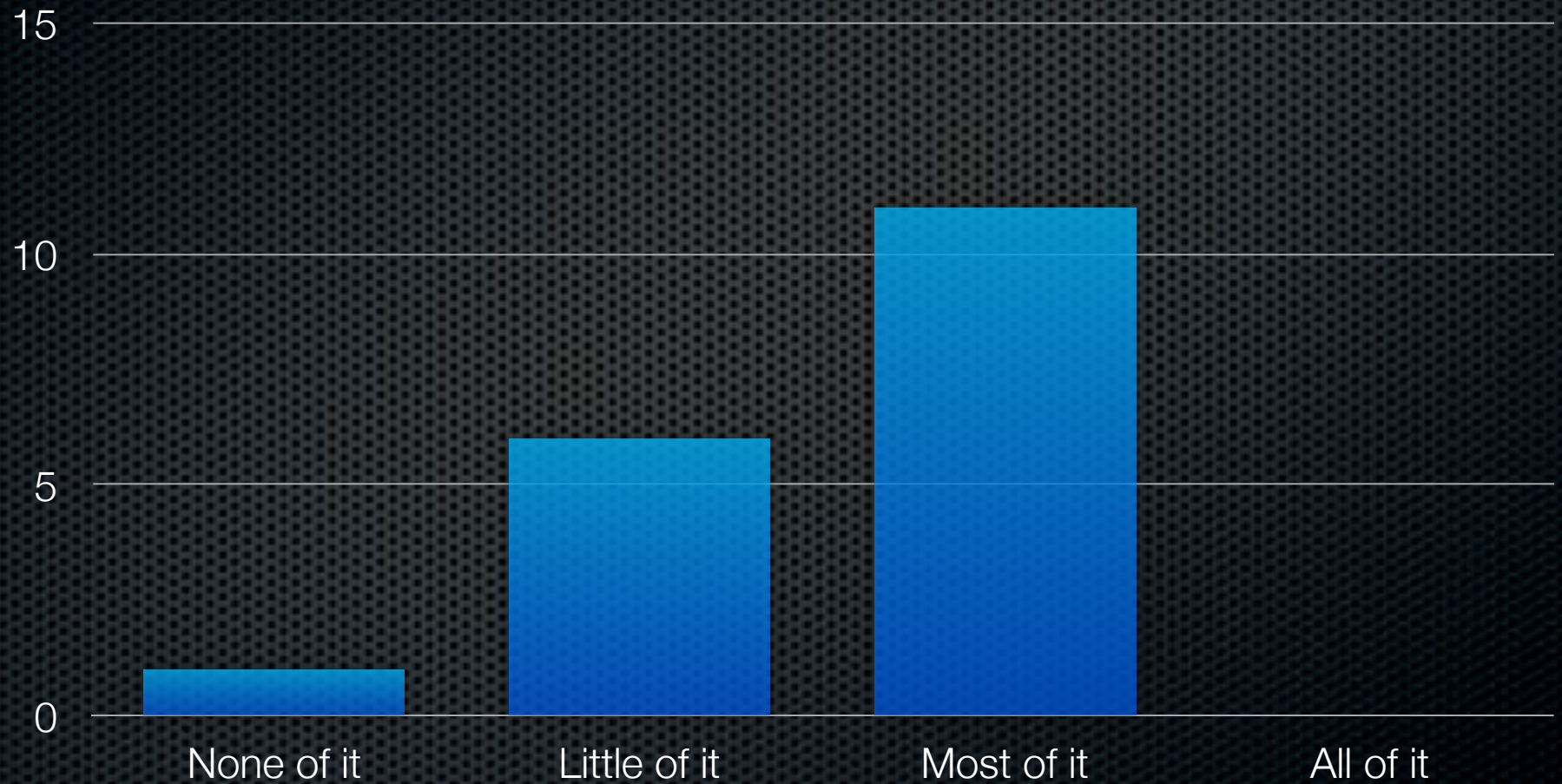
+ Sync with lecture (3)

+ Diagrams (2)

∂ Add random, get-to-know-you questions (3)

∂ Longer questions (2)

# Reading Done

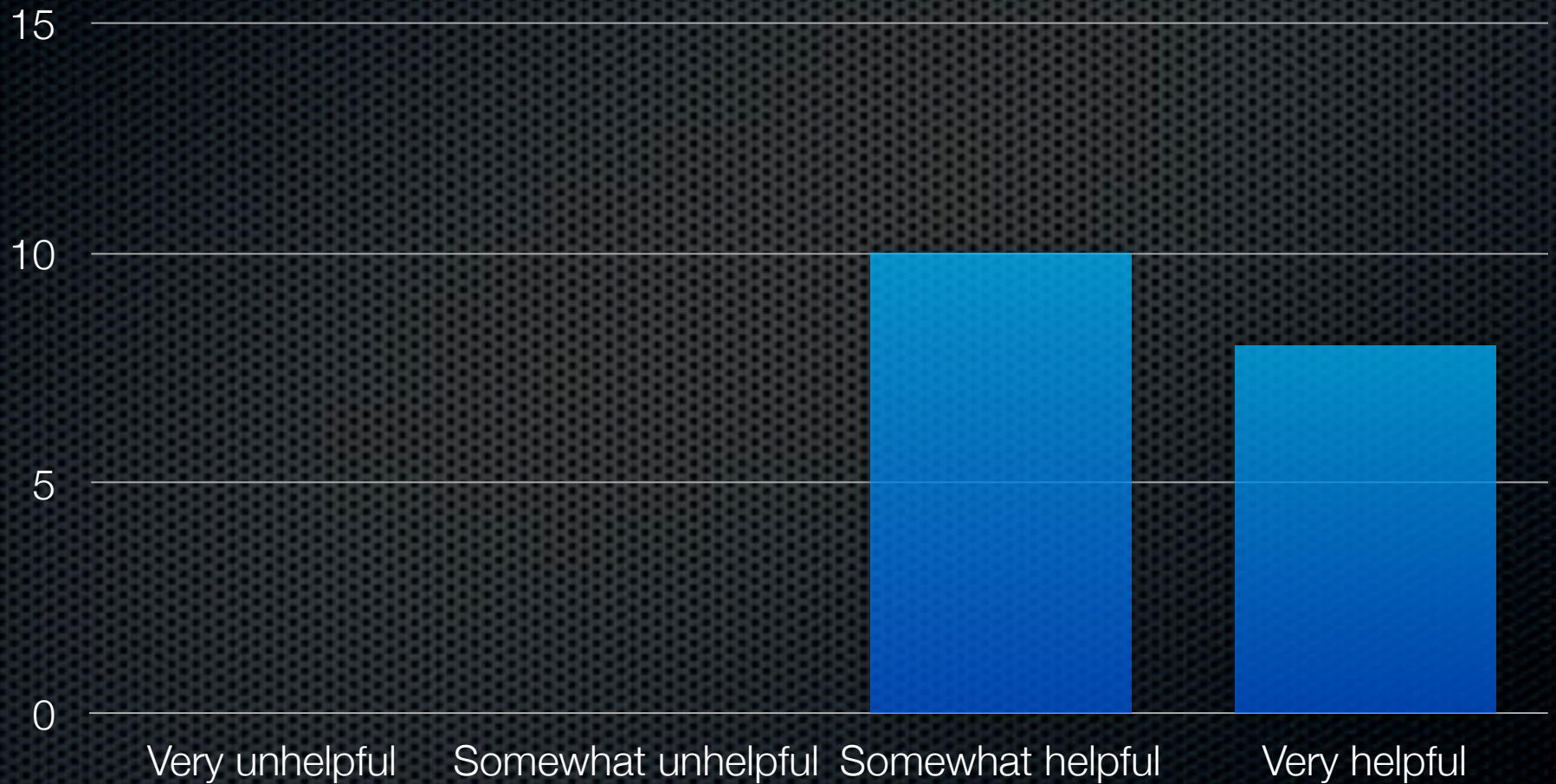




# Homework Difficulty



# Homework Helpfulness



# Homework +/-

+ Good prep for milestones (6)

+ Reinforce lecture/book (4)

+ Hands on learning (4)

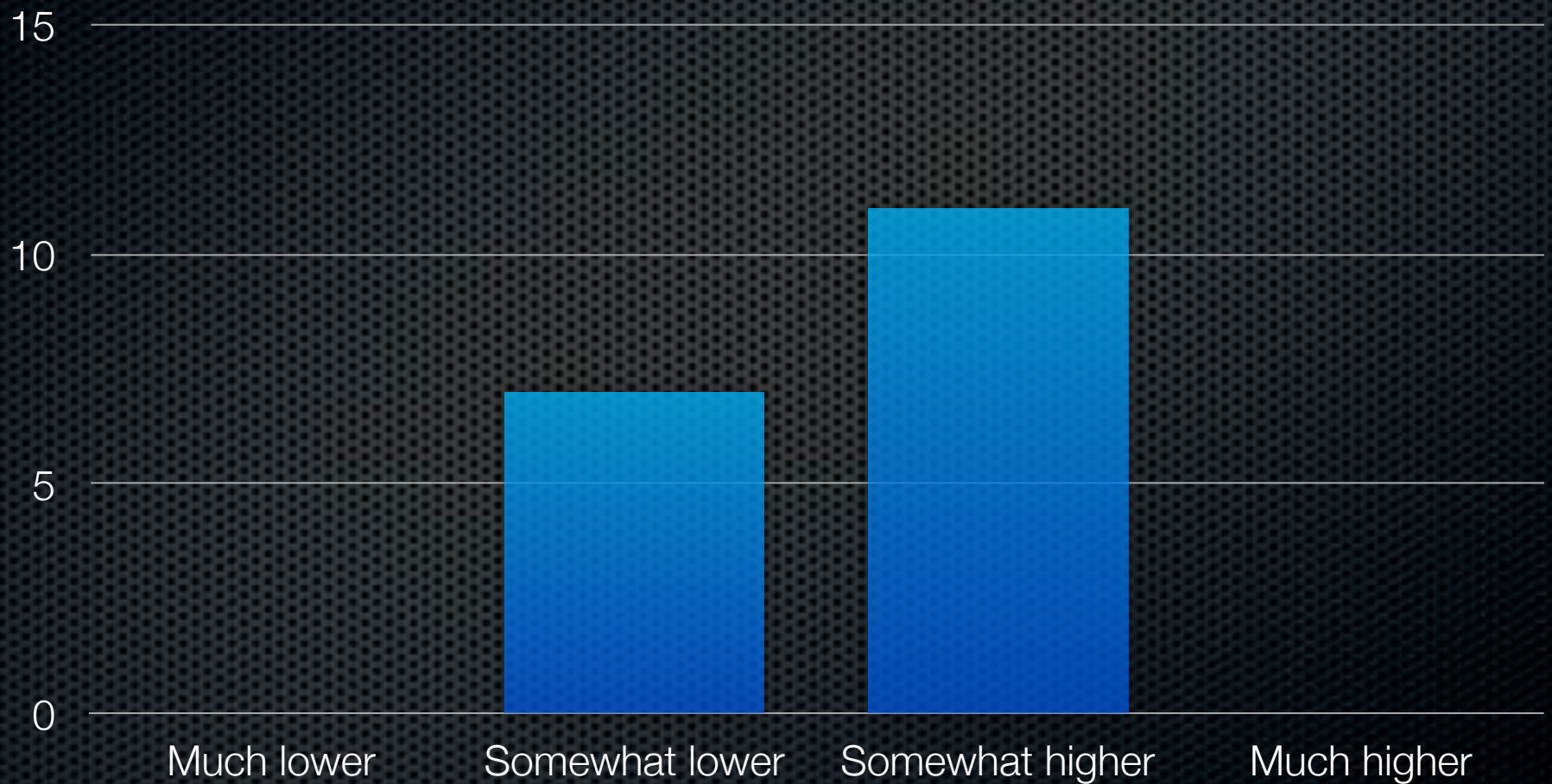
- Too ambiguous (3)

- Provide references to diagram types in the book (2)

- Strange due time (2)

- Break up into smaller parts (2)

# Workload Compared to Average Rose Course



# Other Comments

- ✦ Although workload was rated lower, it seems to be picking up. Go easy. (3)
- ✦ Argh, give us an “about right” option! (1)

# Mastering Object-Oriented Design

- ✦ A **large set** of **soft principles**
- ✦ It isn't magic. We learn it with:
  - ✦ Patterns (named, explained, and applied)
  - ✦ Examples
  - ✦ Practice

“The critical design tool for software development is a mind well educated in design principles.”

# Responsibility Driven Design

# Responsibilities for an Object

- ✦ Doing
  - ✦ a *Sale* is responsible for creating instances of *SalesLineItem*
- ✦ Knowing
  - ✦ a *Sale* is responsible for knowing its *total* cost



# Responsibilities Come in All Sizes

- Big: provide access to a relational database
- Small: create a Sale

A responsibility is **not** the same thing as a method

# Responsibility Driven Design is a Metaphor

- Think of objects as people with responsibilities
- They collaborate to get things done
- An OO program is a “community of collaborating ... objects”

Who owns  $x$ ?

# When Do We Assign Responsibilities?

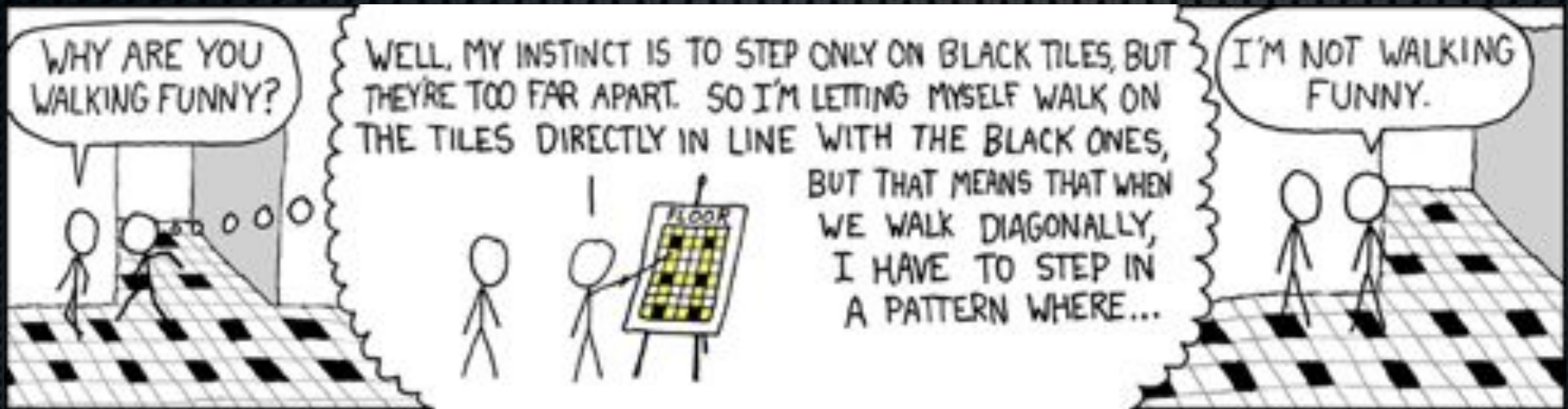
- ✦ While coding
- ✦ While modeling
  - ✦ UML is a low-cost modeling tool
  - ✦ Can assign responsibilities with minimal investment



# GRASP

- A **set of patterns** for **assigning responsibilities** to **software objects**

# Floor Tiles



<http://xkcd.com/245/>

The worst part is when sidewalk cracks are out-of-sync with your natural stride.

# Patterns

A **pattern** is a *named* and *well-known* problem-solution pair that can be applied in a new context

# Example Pattern

Names Matter!

Pattern Name	Information Expert
Problem	What is a basic principle by which to assign responsibilities to objects?
Solution	Assign a responsibility to the class that has the information needed to fulfill it.

“New pattern” is an oxymoron!

Q5-7



# History

- *A Pattern Language: Towns, Buildings, Construction*  
Alexander, Ishikawa, and Silverstein
- Kent Beck, Ward Cunningham at Tektronik
- *Design Patterns: Elements of Reusable Object-Oriented Software* Gamma, Helm, Johnson, Vlissides

The Gang of Four