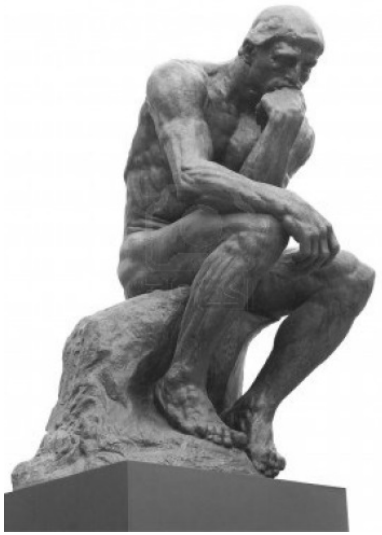
# CS251
# Intermediate Software Design

**Douglas C. Schmidt & Will Otte**
d.schmidt@vanderbilt.edu &
wotte@isis.vanderbilt.edu
www.dre.vanderbilt.edu/~schmidt/cs251

# CS 251 Course Philosophy

- Good design & programming techniques & practices are not best learned through generalities & platitudes
- Instead, it's more effective to see *by example* how significant programs can be made
  - *easier* to write & read,
  - *easier* to maintain & modify, &
  - *more* efficient & resilient

  via the application of time-proven software patterns & advanced design/programming techniques

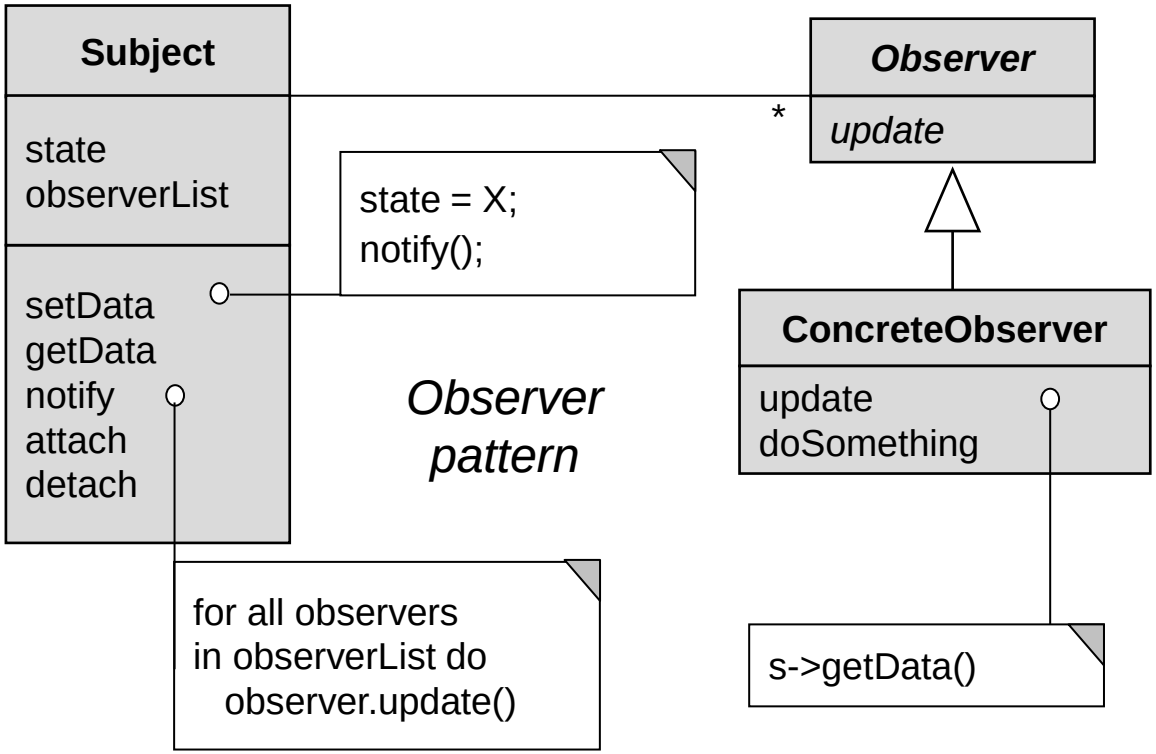


# Summary of CS 251 Course Contents

- Focus on topics important to developing & maintaining quality software:
  - Reuse of patterns & software components
  - Developing, documenting, testing, & applying reusable classes & object-oriented frameworks

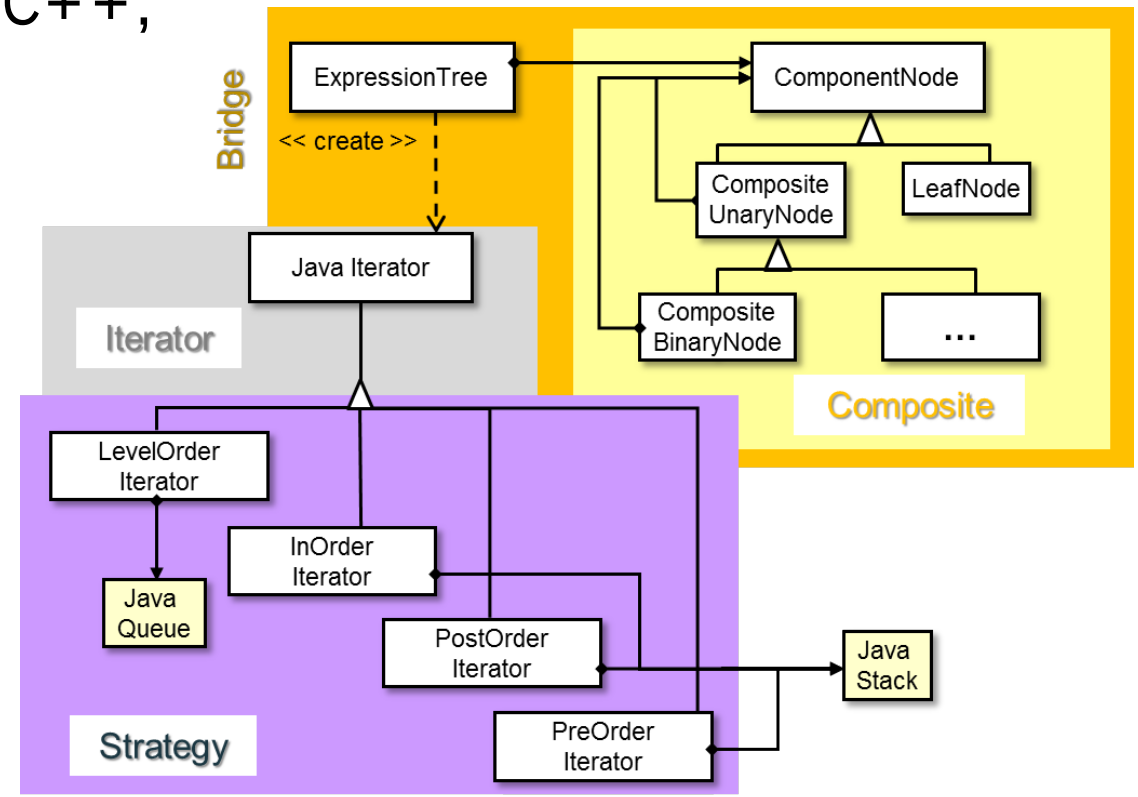
# Summary of CS 251 Course Contents

- Patterns will be taught via an extended case study to provide good role models for software designs & to clearly articulate design tradeoffs



# Summary of CS 251 Course Contents

- Object-oriented techniques will be taught to show how to build software architectures that minimize dependencies & coupling between components
  - We assume you know C++, equivalent to what is covered in CS 201



# Summary of CS 251 Course Contents

- No matter what you know about C++, however, you'll learn a lot more by the time this class is done!
  - Especially C++11, STL, & patterns

# CS 251 Course Work

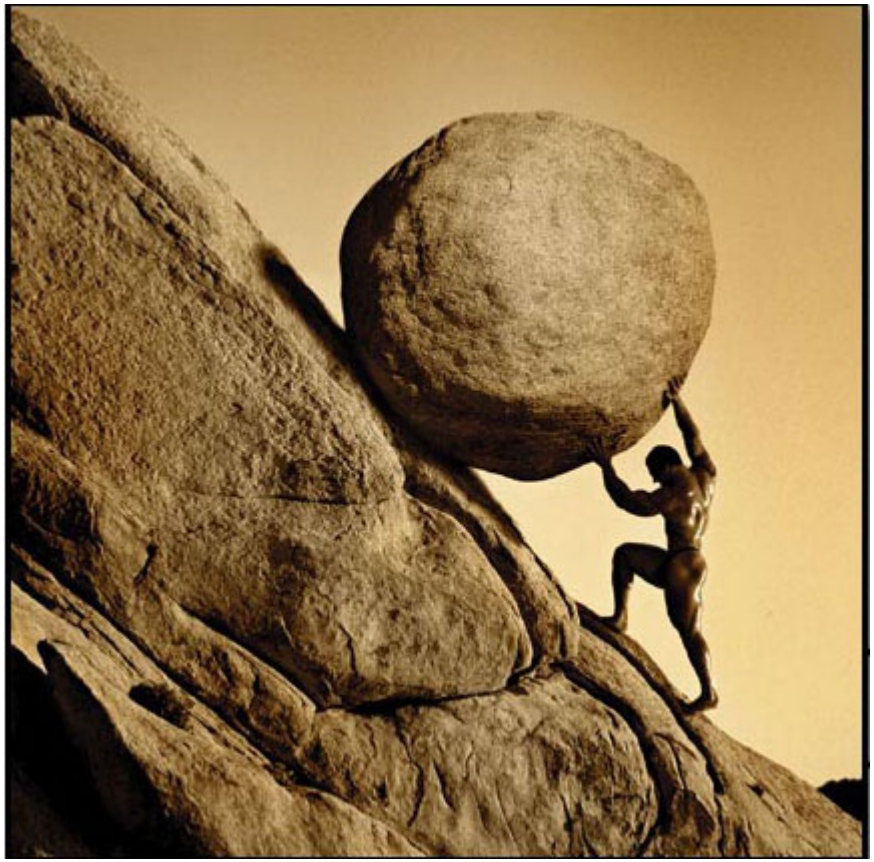
- Programming assignments
  - All programs will be written in C++ & C++11
  - Will be graded using a GNU C++ compiler
    - We provide you with a VM (see course website for details)
  - Must be done individually

# CS 251 Course Work

- Programs will be graded based on:
  - 40% Execution correctness
  - 30% Structure (e.g., modularization, information hiding, etc.)
  - 10% Insightful programming (e.g., developing reusable class components, etc.)
    - 5% effort-based assessment at first submission
  - 10% Consistent style (e.g., capitalization, indenting, etc.)
  - 10% Appropriate commenting style

# CS 251 Course Work

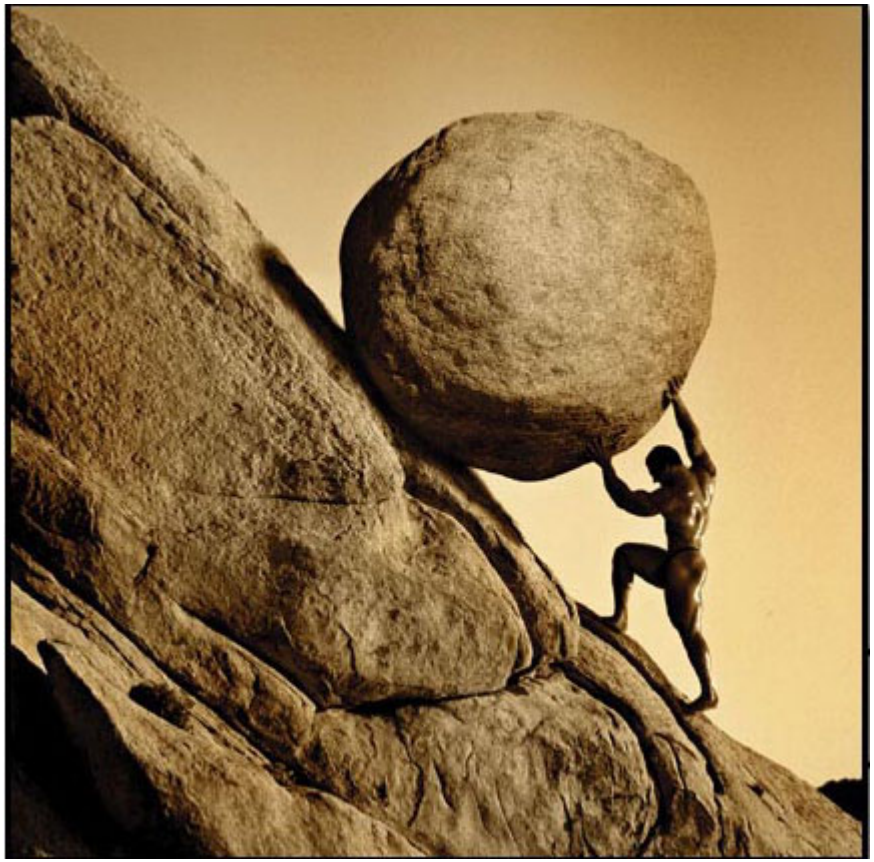
- Programs turned in after the due date will receive a 0
- Weekly quizzes
  - Starting next week on Wednesday
- A Final Exam



# CS 251 Course Work

- The relative weighting* of each portion of the course is:
  - 45% Quizzes
  - 40% Programming projects
  - 10% Final exam
  - 05% Participation

* Relative weighting may change



# CS 251 Semester Outline

- **C++ overview/review**
  - History & evolution
  - Summary of programming paradigms & how they can be realized in C++
- **Advanced C++ programming**
  - Generic programming, exception safety, & memory management
  - C++11 features
    - e.g., range-based for loops, Lambda expressions, move semantics, type traits, etc.
- **C++ Standard Template Library**
  - Containers, iterators, algorithms, etc
- **Gang-of-Four design patterns**
  - Expression tree case study

# CS 251 Ground Rules

- Assignments *must* be submitted on time
- Work *must* be your own
  - www.vanderbilt.edu/student_handbook/the-honor-system#statement-of-the-honor-code
- *No* laptops open, texting, iPads, or smartphones during lecture or quizze
  - Failure to comply will affect your cl participation grade..
- Audio versions of the lectures (plus slides) will be recorded & placed on the course website after class

# CS 251 Ground Rules

- You may be called upon periodically to answer questions
- You'll get out of this course what you put into it, so be prepared to work hard
- Be prepared for quizzes, lots of in class discussions, many hours of programming assignments, & occasional guest lectures
- Make *sure* to avail yourself of available help
  - e.g., office hours, TAs, Piazza, email, recorded lectures, etc.

# CS251 Office Hours & Evacuation Plan

- All office hours will take place in Featheringill Hall room 226
- Check course website www.dre.vanderbilt.edu/~schmidt/cs251 for the list of office hour times
- See engineering.vanderbilt.edu/about/evacuationplans.php for the class evacuation plan