Design patterns
Admin

- Final exam time
- Milestone 2 deadline tomorrow!
Design Patterns

Elements of Reusable Object-Oriented Software

Erich Gamma
Richard Helm
Ralph Johnson
John Vlissides

Foreword by Grady Booch
What are design patterns

- Solutions to specific problems in OO software design
- 23 patterns in 3 categories
  - Creational
  - Structural
    - Composite
    - ...
  - Behavioral
    - Observer
    - Interpreter
    - ...
Observer

• One to many relationship
  • The many need to know changes in “one” immediately

• Example
  • Facebook feed
  • Lines & rectangles
  • ...
Example

• If a person changes its status, how to let all his “subscriber” knows?
Example

• What if there are different types of subscribers?
Class diagram

for each view in views
  v.update()
Can you think of some examples?
Composite pattern

- Tree hierarchy
How to build a tree and traverse it?

```c
struct node{
    struct node* left;
    struct node* right;
    int val;
    int sum(){
        ...
    }
}
```
How to differentiate leaves and others?

```c
struct leaf{
    int val;
    int sum(){ return val;}
}
```
How to accommodate different types of internal nodes?

• Examples
  • struct node or struct leaf?
  • Book
  • Graphics
// Container functionality:
// for each element
elements[i].doThis();
Can you think of some examples?
Interpreter

• What is an interpreter
  • Language, compiler

• Example
  • Boolean expression
    • Abstract syntax tree

    a && b || !c
    a parser will turn this into an abstract syntax tree, and then an interpreter will evaluate the tree. How to write a program to do the tree-based evaluation?
Class diagram

- Client
  - Context
  - TerminalExpression
    - solve(inout Context)
- AbstractExpression
  - «interface»
  - solve(inout Context)
- CompoundExpression
  - solve(inout Context)

Perform "parent" functionality then delegate to each "child" element
"Context" is data structure for holding input and output