Component
operation

Concrete Component
operation()

Concrete Dec #1
addedState
operation()

Concrete Dec #2
addedBehavior()
operation()

Decorator
operation()

component->operation()

Decorator::operation()
addedBehavior()
This course discusses standard patterns of structure and interaction between classes.

- Standard *patterns* of structure and interaction between classes
  - *Design patterns*

- How to apply them to your application
  - *Deal with subsystems at the higher level of abstraction provided by the patterns*

- What to do when it does not fit exactly
  - *Evaluate options and analyze the trade-offs*
At the code level you know some standard patterns.

- How do you walk through all the elements in an array in Java?

```java
for (i = 0; i < array.length; i++) {
    // use the array element at array[i]
}
```

// Since Java 1.5
for (Object element: array) {
    // use the array element referenced
    // by the variable element
}
Our level of discussion for this course is a small subsystem of 3 to 10 classes.

- Higher than what we've done before
  - *Not specific data structures*
  - *Not algorithmic approaches*

- Lower than whole architectures or frameworks
  - *Not financial systems*
  - *Not air-traffic control*
  - *Not Django or Spring framework*
This course uses a problem-based learning methodology.

- Solving problems motivates your learning
- Much of what you learn will not come from listening to a lecture
- This is better because
  - *Learner actively engages the material*
  - *Deeper learning when learner motivates need for knowledge*
  - *More closely resembles true career situation*
Other than not wanting an active experience, the instructor can help you overcome PBL negatives.

- Negatives expressed or perceived by students
  - *Thinking is hard*
  - *Making mistakes during discovery is discouraging*
  - *Not a passive sport anymore*
    - Identify needed knowledge
    - Initiate requests for additional guidance
  - *Don’t know enough to know what I don’t know*
  - *Not getting money’s worth from the instructor*
Success in this course requires a different strategy than for other courses.

- Keep work moving forward
- Seek feedback every class
- Bring up the struggles for discussion
Grading is divided into team and individual components.

<table>
<thead>
<tr>
<th>Component</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exams (10, 10, 25)</td>
<td>45</td>
</tr>
<tr>
<td>Design Problem</td>
<td>5</td>
</tr>
<tr>
<td>Class Activities and Discussion Participation</td>
<td>20</td>
</tr>
<tr>
<td>Design Project (10, 10)</td>
<td>20</td>
</tr>
<tr>
<td>Refactoring Project</td>
<td>10</td>
</tr>
</tbody>
</table>