Problem Decomposition Revisited
(Again):
Object Oriented Design

There’s more...?
Outline

- **Object Oriented Design**
  - Identify the Classes
  - Identify what Information each Class Needs
  - Identify what each Class Needs to Do
Software Development Life Cycle

1. Understand the Problem = Requirements Analysis
2. Work out the Logic = Design
3. Convert it to Code = Implementation
4. Test/Debug
5. Maintenance

Today we will talk about requirements analysis and object oriented design.
You have been hired to automate bank operations for a local credit union. They have told you that their business operates as follows:

- Customers can open accounts. They can make deposits and withdrawals and can close accounts also. On some accounts interest needs to be added, and sometimes fees are deducted.
- All employees can help customers with deposits and withdrawals. Only some employees are authorized to open and close accounts.
Summary

- **Object Oriented Design**
  - Identify the classes
  - Identify what information each class needs
  - Identify what each class needs to do
  - Identify use cases
  - Define the API
  - Define the instance variables
  - Finally – write some code!