Outline

• Object-oriented design
• Identifying Objects
• Initial Approach
Object-Oriented Design

• Identify objects and classes
• Identify how objects interact in system
• Identify hierarchies of related classes
Identifying Objects and Classes

- Requires experience to do really well
- Most guidelines will miss some
- Use several approaches together
- Better to find too many than too few
- Use given approach to gain experience, add more techniques later
Initial Approach

1. Find nouns in requirements
2. Find potential objects from problem domain
3. Look for kinds of things to be modeled
4. Eliminate false objects
Finding Nouns

• Study features of system
• Look for nouns (people, places, things)
• Example features:
  – “add course grade to student record”
  – “enter rental equipment description”
  – “add frequent flyer miles to customer record”
Problem Domain Objects

- Identify “potential objects” through discussions with domain expert
- Domain expert is person that works in domain in which system will be used
- Goal is to identify objects from the way that users/experts think about problem
- Usually no domain experts for 2704 progs!
Things to be Modeled

• Use classification of kinds of objects
  – Several different ones
  – Pick one that you can remember
• Think about what objects in categories occur in the problem
Categories of Classes

- Tangible things – from problem domain
- System interfaces and devices
- Agents – objects to carry out operations
- Events and Transactions – something done
- Users and roles – a system user
- Sub- or external systems – overall system
- Containers
Eliminate False Objects

• Use definition of object to eliminate

• A (problem domain) object is
  (1) a real world entity that is
  (2) important to the discussion of the
  requirements, and
  (3) has a crisply defined boundary