# CS630 Study Guide for Test 1

## r e  terms / concepts

**requirements engineering:**
• People, Procedures, Software, Hardware, Data, Networks
• RE's role as risk management
• Requirement abstraction levels: business model, business process, IS
• Requirement documentation: roles, content, management
• C.A.S.E. support for RE
• Brook's accidental vs. essential difficulties in system development
• Requirement elicitation and mutual client / analyst issues of trust
• Prototyping as elicitation
• Scope / Risk assessment: migrate, automate, integrate, innovate, reengineer, maintain
• RE roles:  domain expert, end-user, requirements engineer, software engineer, project manager, client
• RE and Modeling: identification, description, communication, explanation
• RE project management; RE life cycle
• Requirements process improvement: validation, verification
• Requirement traceability

Testing in the RE section will consist of discussion questions intended to draw on the breadth of your understanding of the components and issues in requirements engineering.

## modeling

### complexity management

Be prepared to define and explain the tools for managing abstraction:

Abstraction (procedural vs data), Encapsulation, Inheritance, Association, Communication with Messages, Whole and parts, Classes and members, Scale, Categories of behavior: immediate causation, change over time, similarity of functions

### object modeling ontology

Be prepared to recognize the tools for describing systems in the object-oriented paradigm:

• object - (identity, encapsulation)
• attribute
• data - (remembrance)
   • static
      • data attribute variable
   • dynamic
      • data attribute value
• behavioral
   • static
      • service
   • dynamic
      • method (operation)
• class - (instance, membership OF)
• relationships
• structural
   • inheritance - (override, parent class / child class, class hierarchy)
• behavioral
   • association - (composition, membership IN)
   • message passing - (sender, receiver, message, parameters)
   • polymorphism - (binding)

### object modeling with UML

Be prepared to read and interpret or to draw UML class diagrams accompanying text model part descriptions.

The test will consist of a combination of any or all of the following: matching of terms and definitions, true / false questions, multiple-choice and short essays.  Questions may include diagram fragments to test your knowledge of diagram syntax and meaning. Seventy-five minutes will be allotted for the test.

Les Waguespack, Ph.D.