Introduction to Modeling

CS 2720
What is a Model?

A *model* is an abstract representation of a system.

Models are well-suited for several important activities, including:

- Communicating with a customer
- Verifying a system *before* building its entirety
- Visualizing interactions and relationships
- Reduce complexity
Domain vs. Design Modeling

We can use models to describe the context our system will be deployed in (domain modeling), or to describe how our actual system will be built (design modeling).

For instance, consider the landscape company scheduling system we discussed earlier. What additional information might we want to know?
We will learn a little bit about the *Unified Modeling Language* (UML). We’ll learn basic notations to ensure *diagrams* convey the correct meaning of the *model*.

Over-simplified, you can think of the diagram as the “syntax” and the model as the “semantics”.
UML defines over a dozen diagrams to represent different perspectives of the model.
We will focus on four of the UML diagrams, covering at least one from each “perspective”:

- Class
- Use Case
- State Machine
- Sequence