

CS 330 is about
Models for CS, but ...

- **What is a model?**
- **Why does CS need them?**

How do Models and CS relate to ...

- **Abstraction**
- **Generalization**
- **Principles**
- **Theory**
- **Precision**
- **Structure**
- **Concepts**
- **Thought**

Models and Good Ones

Model: a collection of precisely stated interacting ideas that focus on a particular aspect of a

- **thing**
- **topic**
- **situation**
- **class of problems**

A good model should simplify a topic to its essence, stripping away the details, so we can reason precisely about it.

- **frictionless plane**
- **elliptical orbits**
- **organic molecule models**
- **supply and demand curves**

A formal model is expressed in terms of symbols, with precise and consistent rules about how to manipulate them, to prove things.

Why Models?

Conventional wisdom is to . .

- **Reuse ideas and software**
- **Learn from mistakes / experience / history**

But, situations differ, so . .

- **Use models to capture similarities**

Models and theory . .

- **Do not deny experience, but capture its essence**

How to Model

- **Ignore some parts.**
- **Ignore some processes.**
- **Keep the key parts and processes.**
- **Keep what you're interested in.**

Models of What?

- **Computers: architecture, not digital electronics**
- **Language: language concepts, not one language**
 - ✓ **Arrangement of symbols**
 - ✓ **Meanings of symbols and arrangements**
- **Software: software engineering, not one application**

Models for Computer Science

Logic (used in CS for)

- **Software Engineering**
- **Artificial Intelligence**
- **Databases**

Language Models (used in CS for)

- **Compilers**
- **Human Language Technology**
- **Computability Theory**