Introduction to Specification Languages

CS 2720
How are requirements kept in the two general processes we have seen?

What *language* is used for these requirements?
One established “academic” approach to specification that has recently gained a lot of traction in “industry” is the use of a specification language.

A specification language is a formal language in computer science used during systems analysis, requirements analysis and systems design to describe a system at a much higher level than a programming language, which is used to produce the executable code for a system. (Specification language: Wikipedia)
Why Specification Languages?

From *Specifying Systems*:

- Mathematics is nature’s way of letting you know how sloppy your writing is.
- Formal mathematics is nature’s way of letting you know how sloppy your mathematics is.
- A system specification consists of a lot of ordinary mathematics glued together with a tiny bit of temporal logic.
- Fortunately, the math is quite simple. If exposure to C++ hasn’t destroyed your ability to think logically, you should have no trouble filling any gaps in your mathematics education.
Examples of Specification Languages

- **Z** (introduced in 1977)
- **Alloy** (introduced in 1997)
- **TLA+** (introduced in 1999)
Some Fun Demos!

You, too, can enjoy the fun that is formal specification!!

- CZT: Community Z Tools
- Alloy
- TLA Toolbox
To help see what a formal specification looks like, let’s consider the following problem:

You are tasked with building a system to manage course registration. The system should track the room and meeting times of the course, the course’s instructor, and the students registered for it.