

# Choosing a Representation Language

You need to represent a problem to solve it on a computer.

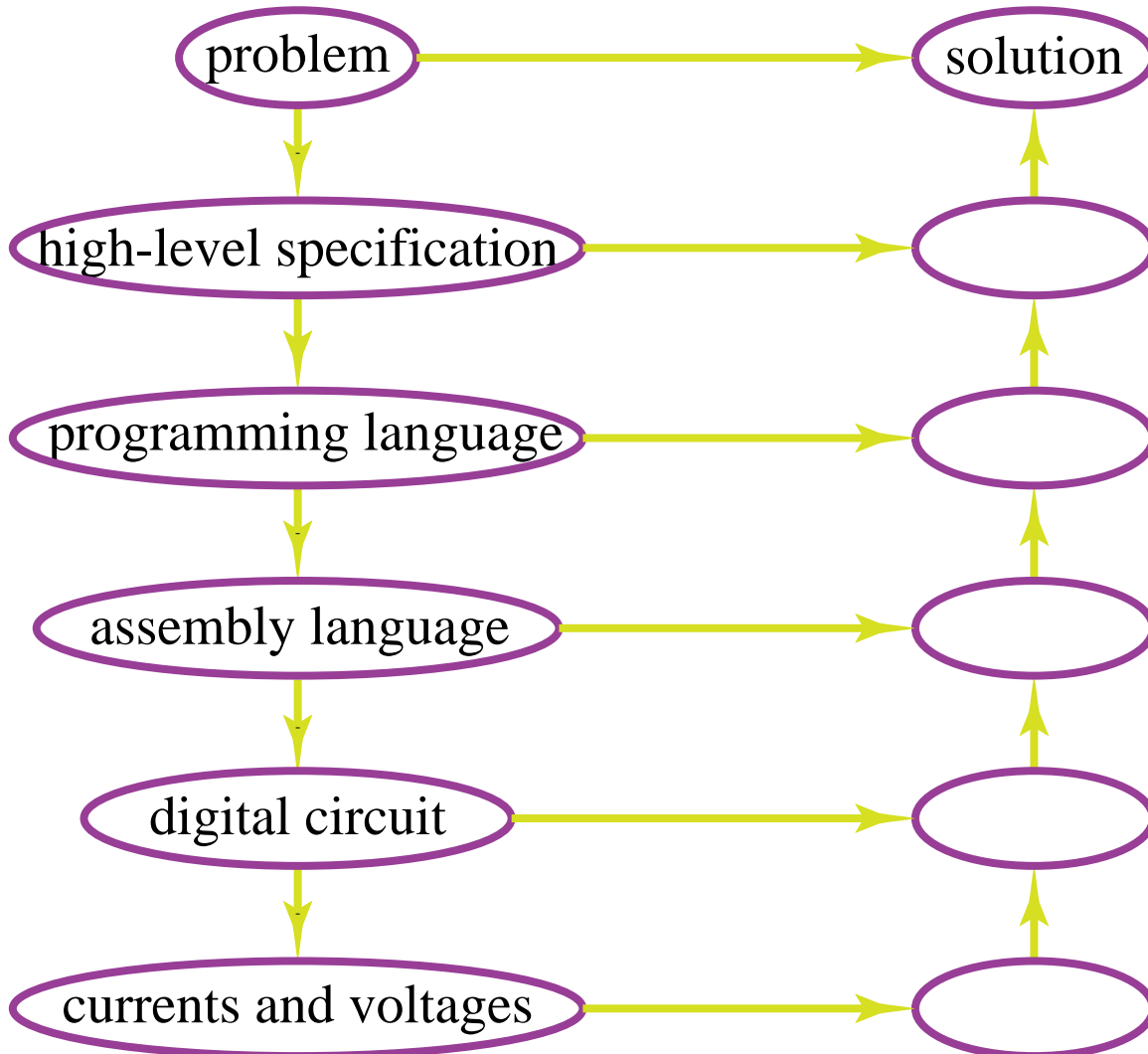


Example representations: C++, CILog/Prolog, English

A **logic** is a language + specification of what follows from input in that language.



# Hierarchy of representations



# Knowledge & Symbol Levels

Two levels of abstraction seem to be common among biological and computational entities:

**Knowledge level** in terms of what an agent knows and what an agent's goals are

**Symbol level** in terms of what symbols the agent is manipulating.

The knowledge level is about the external world to the agent.

The symbol level is about what symbols an agent uses to implement the knowledge level.



# Mapping from Problem to Representation

- What level of abstraction of the problem do you want to have to represent?
- What objects and relations in the world do you want to represent?
- How can you represent the knowledge to ensure that the representation is natural, modular, and maintainable?



# Choosing a level of abstraction

- A high-level description is easier for a human to specify and understand.
- A low-level description can be more accurate and more predictive. High-level descriptions abstract away details that may be important for actually solving the problem.
- The lower the level, the more difficult it is to reason with.
- You may not know the information needed for a low-level description.

It is sometime possible to use multiple levels of abstraction.



# Choosing Objects and Relations

How to represent: “Pen #7 is red.”

*red(pen<sub>7</sub>)*. It’s easy to ask “What’s red?”

Can’t ask “what is the color of *pen<sub>7</sub>*?”

*color(pen<sub>7</sub>, red)*. It’s easy to ask “What’s red?”

It’s easy to ask “What is the color of *pen<sub>7</sub>*?”

Can’t ask “What property of *pen<sub>7</sub>* has value *red*?”

*prop(pen<sub>7</sub>, color, red)*. It’s easy to ask all these questions.

*prop(Object, Attribute, Value)* is the only relation needed:

**object-attribute-value representation**



# Universality of *prop*

To represent “a is a parcel”

$prop(a, is\_a, parcel)$ , where  $is\_a$  is a special relation

$prop(a, parcel, true)$ , where  $parcel$  is a Boolean attribute

To represent  $scheduled(cs422, 2, 1030, cc208)$ . “section 2 of course  $cs422$  is scheduled at 10:30 in room  $cc208$ .” Let  $b123$  name the booking:

$prop(b123, course, cs422)$ .

$prop(b123, section, 2)$ .

$prop(b123, time, 1030)$ .

$prop(b123, room, cc208)$ .

