Decision vs Optimization

In algorithms, we are usually concerned with optimization.
- Shortest path
- Minimum spanning tree
- Maximum flow

There is a corresponding decision problem variant.
- Does G have a path of length at most k.
- Does G have a MST of weight at most k.
- Does G have a flow of at least k.

Decision problems can be simple yes/no questions.
- SAT
- Ham - Path
- Halting language

Formal Languages

An alphabet Σ is a finite set of symbols.

A string over an alphabet Σ is a finite sequence of symbols in Σ.

Ex) $\Sigma = \{0, 1\}$ 01101   1   000000

The set of all strings over an alphabet Σ is Σ^*.

Ex) $\{0,1\}^* = \{\varepsilon/\lambda, 0, 1, 00, 01, 10, 11, 000, \dots\}$

The empty string is ε (or λ).

The set of all strings over Σ except ε is Σ^+.

A language over Σ is a subset of Σ^*.

We can encode decision problems as languages.
A yes/no decision corresponds to in/out of the language.

Ex] Given x, y, z, does $x + y = z$?

$$\Sigma = \mathbb{Z}_{10} \cup \{+, =\}$$
$$ADD = \{ x + y = z \mid x, y, z \in \mathbb{Z}_{10}^+ \text{ and } x + y = z\}.$$

Usually, the alphabet we use is $\{0, 1\}$. Any alphabet
we might choose can be encoded into binary strings
with only a constant factor slowdown.