Computational Problem Solving

In order to solve a problem computationally, two things are needed:

1. A representation that captures all the relevant aspects of the problem, and
2. an algorithm that solves the problem by use of the representation.

Let's consider a problem known as the Man, Cabbage, Goat, Wolf problem:

A man lives on the east side of a river. He wishes to bring a cabbage, a goat, and a wolf to a village on the west side of the river to sell. However, his boat is only big enough to hold himself, and either the cabbage, goat, or wolf. In addition, the man cannot leave the goat alone with the cabbage because the goat will eat the cabbage, and he cannot leave the wolf alone with the goat because the wolf will eat the goat. **How does the man solve his problem?**

There is a simple algorithmic approach for solving this problem by simply trying all possible combinations of items that may be rowed back and forth across the river. **Trying all possible solutions to a given problem is referred to as a *brute force approach*.** What would be an appropriate representation for this problem? Since only the relevant aspects of the problem need to be represented, all the irrelevant details can be omitted. **A representation that leaves out details of what is being represented is a form of *abstraction*.**

The use of abstraction is prevalent in computer science. In this case, is the color of the boat relevant? The width of the river? The name of the man? No, the only relevant information is where each item is at each step. The collective location of each item, in this case, refers to the *state* of the problem.

Thus, the *start state* of the problem can be represented as follows:

man cabbage goat wolf

[E, E, E, E]

In this representation, the symbol E denotes that each corresponding object is on the east side of the river. If the man were to row the goat across with him, for example, then the representation of the new problem *state* would be:

[W, E, W, E]

in which the symbol W indicates that the corresponding object is on the west side of the river—in this case, the man and goat. (The locations of the cabbage and wolf are left unchanged.) A solution to this problem is a sequence of steps that converts the initial state,

[E, E, E, E]

in which all objects are on the east side of the river, to the goal state,

[W, W, W, W]

in which all objects are on the west side of the river. Each step corresponds to the man rowing a particular object across the river (or the man rowing alone).

In order to solve a problem computationally, two things are needed: a representation that captures all the relevant aspects of the problem, and an algorithm that solves the problem by use of the representation.