Chapter 6

Series of Choices

The divide-and-conquer approach solves problems via decomposition: problems are broken up into smaller instances, solved, and the subproblem solutions combined. A contrasting approach is to build up the solution incrementally by making a series of choices or decisions.

Key to this approach is that series of choices or decision is the same kind of choice or decision made about each in a series of elements, not a sequence of choices about different kinds of things.

6.1 Series-of-Choices Paradigms

Each choice in the series of choices involves picking one of one or more alternatives. How many alternatives must be considered to find a solution is a fundamental characteristic of the problem and leads to several different algorithmic paradigms.

- If only **one** alternative needs to be considered, the formulation can be iterative. The key focus for the algorithm is determining how to pick that right alternative for each decision, and showing that the series of choices made leads to a correct solution. Greedy algorithms are of this type and will be considered in chapter 7.
- If **more than one** alternative needs to be considered, the formulation is typically recursive. The key focus for the algorithm is how to avoid an exponential blowup in running time. Backtracking, branch-and-bound, and dynamic programming algorithms are of this type and will be considered in chapters 8 and 9.

6.2 Series-of-Choices Patterns

Independent of the number of alternatives that must be considered, series-of-choices algorithms can be categorized in several ways. One is by what the series of choices is about:

- **process input**, where the choice or decision is about what to do with each input element in turn
- **produce output**, where the choice or decision is about what what the next output element is

Another characterization is according to the type of task. Common categories include:

- a **subset**, where the task to is select a subset of the input items subject to some subset membership constraint
- a **sequence**, where the task is to produce an ordering of all or a subset of the input items
- a **labeling**, where the task is to assign labels to the input items

6.3 Optimization Problems

The goal of an optimization problem is to find the best solution (as defined by some measure) amongst (generally many) valid solutions. It is useful for optimization problems to distinguish between a *legal solution* and an *optimal solution*.

When using a series-of-choice approach for an optimization problem, the goal is make the right choice — picking the alternative that leads to an optimal solution — for each decision.

6.4 Elements of Series-of-Choices Algorithms

6.4.1 The Series of Choices

The patterns in section 6.2 give rise to different variations for the series of choices:

task	iterative pattern	choice
subset	process input	include element in the solution or not
	produce output	which element to include in the solution next
ordering	process input	where to add the element into the solution-so-far
	produce output	which element to append to the solution-so-far
labeling	process input	the label to give the element

It can be easier to define how the choice is made (which alternative is picked) for the produce output patterns. Note that for subset tasks, the process input and produce output approaches are directly related — if the elements are sorted according to the choice criterion, then "which element to include in the solution next" reduces to asking "include in the solution or not?" until the answer is yes.