San Skulrattanakulchai
Gustavus Adolphus College
Oct 10, 2014

Rod Cutting

CLRS: Ch 15.1

Problem Given a steel rod of length n, and a table of prices $p_i, p_2, \ldots, p_n$ of rods of length $1, 2, \ldots, n$, we want to cut the given rod into pieces of integral lengths so as to maximize the revenue.

Dynamic Programming Solution
For $1 \leq i \leq n$, let r_i be the maximum revenue obtainable from cutting up the rod of length i into integral-length pieces and selling them.
We seek r_n.

Optimal Substructure Property
Consider an optimal cutting of a given steel rod of length i. We'll call the rods resulting from the cutting from left to right as rods 1, 2, ..., etc. Also assume that we make the cuts from left to right. Let rod 1 in this optimal cutting have length k. After the first cut, cutting the remaining rod must be done in an optimal way! We thus have $r_i = p_k + r_{i-k}$. We don't know the value of k but we know k satisfies $1 \leq k \leq i$. This gives us the following recurrence.

Recurrence

$$r_i = \begin{cases} 0 & \text{if } i = 0 \\ \max\{\, p_k + r_{i-k} : 1 \leq k \leq i \,\} & \text{if } 1 \leq i \leq n \end{cases}$$

We seek r_n.

Running Time
The running time is $O(n^2)$.

- Step 1 fills the $R[\cdot]$ table and the mamimizer table. There are $O(n)$ table entries. Each table entry takes $O(n)$ time to compute. So running time is $O(n^2)$ for the table-filling step.

- Step 2 follows each maximizer pointer in $O(1)$ time, $O(n)$ time total.