DP Extra

Minimum Coinage Problem (MCP)

- Given an array C of coin denominations and a number n, find the minimum number of coins to amount to n
  - Simplifications: C must contain 1
  - Unlimited supply of each coin type
- Greedy does not work in general
- E.g., C = [1, 5, 7, 9], n = 13
  - Greedy: 9, 1, 1, 1, 1
  - Optimal: 7, 5, 1

Dynamic Programming

- Optimal substructure → work out the recurrence relation
- Overlapping substructure → work out the dependency

Maximum Subarray Sum Problem (MSS)

- Given an array x of n numbers, find its contiguous subarray such that its sum is the maximum.
- E.g., xs = [-2, 5, -1, 2, -4, 1]
  - MSS = [5, -1, 2]
- What about xs = [-2, 1, -3, 4, -1, 2, 1, -5, 4]?
  - MSS =
- Need an algorithm for non-trivial cases
  - What about a naïve algorithm?

Naïve Algorithm

- Hint: How many (contiguous) subarrays do we have?
  - Enumerate and keep the max
- Each such subarray ←→ (s, e)

Dynamic Programming

- Define max_on[i] be the max subarray that ends on the i-th element
  - If we can compute max_on array, then taking the maximum over it will give the solution for MSS
- max_on[i] = max(max_on[i-1] + x[i], x[i])
- Takes O(n) time and O(1) space

Misc

- The problem can also be solved by
  - Divide-and-conquer
  - Shortest-path
  - Algebraic optimization
  - Read more at: https://en.wikipedia.org/wiki/Maximum_subarray_problem

Misc.

- So far we concentrate on improvements that reduces the complexity in the worst case
- In practice, there are improvements that help reduces the complexity in many cases, but do not help with the worst case complexity.
  - E.g., preprocess the input so that all contiguous positive (resp. negative) numbers were merged into a single 'super' element. If an algorithm works in $O(n^2)$ time, then this may be reduced to $O(m^2)$ time.
    - Does not help with the worst case unless $m = o(n)$

Misc.

- Another famous example: Sunday's algorithm for string matching
  - Let text be a long string, and pattern be a (short) query string. Find one occurrence of pattern in the text (if any)
  - KMP matches pattern to the text from left to right (on the pattern)
  - BM matches from right to the left
  - Sunday matches letters in decreasing order of the frequency of letters in the pattern