

2017-09-25

- **Optimal parenthesization problem:**

- Definition: Given a string of n items (v_1, v_2, \dots, v_n) , find an optimal parenthesization of the string.
- Cost $c(i, j)$:
 - The cost of a parenthesis pair is the sum of the enclosed numbers.
 - The cost of a parenthesization is the sum of the individual costs.
- Example: $(1, 2, 3, 4) \rightarrow (((1, 2), 3), 4)$
- Recurrence: $c^*(i, j) = \min_{i \leq k < j} \{c^*(i, k) + c^*(k + 1, j)\} + \sum_{k=i}^j v_k$
- Complexity: $O(n^3)$

- **Longest common subsequence (LCS) problem:**

- Definition: Given two strings $A = a_1a_2\dots a_n$ and $B = b_1b_2\dots b_m$, find their longest common subsequence.
- Example: $abadc \times adcb \rightarrow adc$
- Recurrence: $L(i, j) = \max\{L(i - 1, j), L(i, j - 1), l_{ij} + L(i - 1, j - 1)\}$
- Complexity: $O(nm)$

- **0/1 knapsack problem:**

- Definition: $\max \sum_{i=1}^n p_i x_i$ subject to $\sum_{i=1}^n w_i x_i \leq M$, $x_i \in \{0, 1\}, i = 1, 2, \dots, n$
- Recurrence: $f(k, g) = \max\{f(k - 1, g), p_k + f(k - 1, g - w_k)\}$
- Complexity: $O(nM)$