CP

▼ DP tricks

1. **1D1D Optimize**
   - $dp[i] = \min\limits_{0 \le j < i} dp[j] + w(j, i)$
   - $w(j, i)$ is cost whice need to satisfies quadrangle inequality:
     - $w(a, c) + w(b, d) \le w(a, d) + w(b, c)$ with $a \le b \le c \le d$
   - Example cost:

     1. $w(j, i) = b[j] \cdot a[i]$, providing that $a[i] < a[j]$ and $b[i] > b[j]$ satisfy for all $i < j$.
     2. $w(j, i) = (x[i] - x[j])^2 + C$, providing that $x[j] < x[i]$ satisfies for all $j < i$.
     3. $w(j, i) = b[j] \cdot a[i] + b[j]^2 \cdot a[i]^2 + b[j]^3 \cdot a[i]^3$, providing that the following is true for all $i < j$: $0 < a[i] < a[j]$ and $b[i] > b[j] > 0$.

2. **Knuth**
   - $dp[i][j] = \min\limits_{i < k < j} dp[i][k] + dp[k][j] + w(j, i)$
   - Call $A(i, j)$ is optimize k give min $dp[i][j]$ then:
     - $A(i, j - 1) \le A(i, j) \le A(i + 1, j)$
3. **Divide and Conquer**
   - $dp[i][k] = \min\limits_{j < i} dp[j][k - 1] + w(j, i)$
   - Call $A(i, k)$ is optimize j give min $dp[i][k]$ then:
     - $A(i, k) \le A(i, k + 1)$
4. **ConvexHull Trick**
5. **AlienTrick**
   - Often combine with above trick in binary search phase.
   - Problem: Use at most K so that the answer is max or min.
   - Call $ans(i)$ is the maximum when using **exactly i then**

- $ans(i+1) - ans(i) <= ans(i) - ans(i-1)$.

6. **SOS**
   - Given a fixed array **A** of 2^N integers, we need to calculate $\forall x$ function $F(x)$ = Sum of all $A[i]$ such that **i** is a subset of **x**.
7. **Open and Close Interval Trick**
8. **BerlekampMassey**

▼ Data Structure

1. **Walking on ST**.
2. **Segment Tree Beast**
   - Solve problem with this type of query:
     - For all $i \in [l, r]$, change A_i to $max(A_i, x)$.
     - For all $i \in [l, r]$, change A_i to $min(A_i, x)$.
     - Get sum of A_i
3. **Splays Tree**
   - http://poj.org/problem?id=3580

▼ Tree

1. **DSU on Tree**
   - Answer how many vertices in the subtree of vertex `v` has some property in time O(nlogn).
2. **HLD**:
   - Update and query on path from vertex u to v.
3. **Compress Tree**:
   - Only consider important node.
4. **DP on Tree**

- O(n^3) seem but O(n^2).
- https://usaco.guide/adv/comb-sub#problem-cf-815C

▼ Others

- Divide and Conquer Tricks when you want to **ignore 1 element**.
  - Problem: https://codeforces.com/gym/104555/problem/D.
- GCD
  - Inclusion - Exclusion : https://codeforces.com/contest/1559/problem/E