SOLUTION 20/10

BÀI 1. TỔNG HÀM SỐ

Subtask 1 (20%): $n \le 80$

Duyệt tất cả các đoạn con liên tiếp $a[l..r]$ và xét mọi i, j $(l \le i \le j \le r)$ để tìm $f(a[l..r])$.

Độ phức tạp thời gian: $\sim O(n^4)$

Subtask 2, 3 (70%): $n \le 5 * 10^3$

Ta giải bằng phương pháp quy hoạch động: (có nhiều cách khác nhau)

Gọi $F[l][r]$ là giá trị $\lceil \frac{|p_j - p_i|}{j - i} \rceil$ $(l \le i < j \le r)$ lớn nhất trong đoạn con $a[l..r]$. Hay nói cách khác: $F[l][r] = f(a[l..r])$.

Cơ sở QHĐ: $F[l][r] = 0$ với $l \le r + 1$.

Với từng độ dài l ta xét từng đoạn $a[i..j]$ $(j = i + l - 1)$ như sau:

- $F[i][j] = \max(F[i][j-1], F[i+1][j])$ lấy giá trị lớn nhất của hai đoạn đã xét trước.
- Nếu $\left\lceil \frac{|a_j - a_i|}{j - i} \right\rceil > F[i][j]$ thì $F[i][j] = \lceil \frac{|p_j - p_i|}{j - i} \rceil$.

Độ phức tạp thời gian: $O(n^2)$ cho khởi tạo.

Nhận xét: Gọi $b_i = |a_i - a_{i+1}|$ $(1 \le i < n)$, ta có thể chứng minh rằng $f(a[l..r])$ là giá trị lớn nhất của dãy $b[l..r-1]$, cách chứng minh như sau:

Gọi X là giá trị lớn nhất của dãy $b[l..r-1]$ và $g(l,r) = \left\lceil \frac{|a_r - a_l|}{r - l} \right\rceil$. Điều cần chứng minh ở đây là $g(l,r) \le X$ với mọi $1 \le l < r \le n$.

Với mọi $1 \le l \le r \le n - 1$, gọi $h(l,r) = \lceil \frac{\sum_{i=l}^{r} b_i}{r - l + 1} \rceil$. Ta có:

$$h(l,r) = \left\lceil \frac{\sum_{i=l}^{r} b_i}{r - l + 1} \right\rceil \le \left\lceil \frac{\sum_{i=l}^{r} X}{r - l + 1} \right\rceil = \left\lceil \frac{(r - l + 1)\, X}{r - l + 1} \right\rceil = X$$

Với mọi $1 \le l < r \le n$, với bất đẳng thức $|a| - |b| \le |a - b|$ ta có:

$$|a_r - a_l| = \left| \sum_{i=l}^{r-1} a_{i+1} - a_i \right| \le \sum_{i=l}^{r-1} |a_{i+1} - a_i| = \sum_{i=l}^{r-1} b_i \Rightarrow \frac{|a_r - a_l|}{r - l} \le \frac{\sum_{i=l}^{r-1} b_i}{r - l}$$

$$\Rightarrow \left\lceil \frac{|a_r - a_l|}{r - l} \right\rceil \le \left\lceil \frac{\sum_{i=l}^{r-1} b_i}{r - l} \right\rceil \Rightarrow g(l,r) \le h(l, r-1) = X$$

⇒ Điều phải chứng minh.

Subtask 4 (100%): $n \le 2 * 10^5$

Với một dãy con bất kì, nếu có nhiều phần tử cùng có giá trị lớn nhất, ta sẽ chọn phần tử có vị trí bé nhất. Với mỗi phần tử b_i, ta cần đếm xem có bao nhiêu dãy con nhận b_i làm phần tử lớn nhất. Ta gọi:

- $L[i]$ là vị trí đầu tiên bên trái vị trí i mà $b_{L[i]} \ge b_i$ (nếu không tìm thấy thì $L[i] = 0$).
- $R[i]$ là vị trí đầu tiên bên phải vị trí i mà $b_{R[i]} > b_i$ (nếu không tìm thấy thì $R[i] = n$).

Khi đó mọi dãy con $b[l..r]$ với $p_i < l \le i \le r < q_i$ sẽ nhận b_i là giá trị lớn nhất. Hay nói cách khác:

$$S = \sum_{i=1}^{n-1} b_i * (i - l_i) * (r_i - i)$$

Độ phức tạp thời gian: $O(n)$.

BÀI 2. DÃY SỐ TỔNG – TÍCH

Subtask 1 (20%): $1 \leq P, S \leq 20, k \leq 5$

Vì các số cần tìm cần là các số nguyên dương và tổng, tích các số liền nhau không vượt quá S nên ta chỉ việc thử các số từ 1 đến S với mỗi $1 \leq i \leq k$.

Độ phức tạp thời gian: $O(S^k)$

Subtask 2, 3 (80%): $P \leq 10^5, S \leq 10^9, k \leq 50$

Gọi $F(i,j)$ là số dãy thỏa mãn khi có i phần tử và kết thúc tại j ($j \leq P, S$).

Công thức QHĐ:

$$F(i,j) = F(i,j) + \sum_{t=1}^{\min\left(\frac{P}{j},\ S-j\right)} F(i-1,t) = F(i,j) + sum\left(i-1, \min\left(\frac{P}{j}, S-j\right)\right)$$

Độ phức tạp thời gian: $O(\sim \min(P,S) * \log(P) * k)$

Subtask 4 (100%): $1 \leq P, S \leq 10^9, k \leq 50$

Nhận xét: Xét 2 số $a \leq b$ sao cho $a \times b \leq P \rightarrow a \times a \leq P \rightarrow a \leq \sqrt{P}$.

Ta lập một mảng V, với mỗi giá trị của a ta sẽ lưu lại $\frac{P}{a}$ và $S - a$. V gồm $3 * \sqrt{P}$ phần tử.

Do đó ta giảm độ phức tạp của bài toán bằng cách đặt $F(i,j)$ là số dãy thoả mãn khi dãy có i phần tử và kết thúc tại giá trị V_j.

Độ phức tạp thời gian: $O(3 * \sqrt{P} * \log(P) * k)$.

BÀI 3. TRẠM PHÁT ĐIỆN

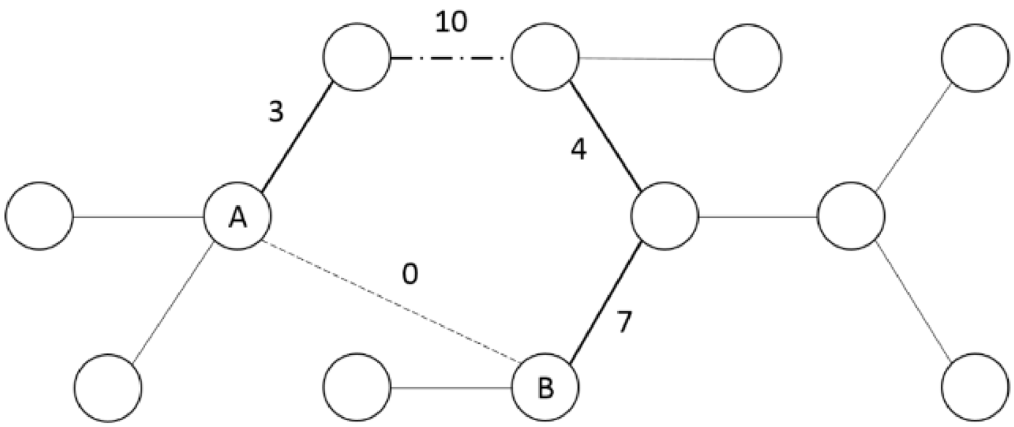
Subtask 1, 2 (35%): $N, M \leq 15, Q \leq 100$ hoặc $N \leq 4 * 10^3, M \leq 4 * 10^5, Q = 1$

Trước hết, với phương án đặt trạm điện tại hai thành phố A và B, ta có thể xem như chỉ đặt trạm điện tại thành phố A, và có thể xây dựng đường dây điện nối giữa A và B với chi phí 0. Từ đó, thuật toán cho hai subtask đầu tiên là bổ sung thêm cạnh $(A, B, 0)$ và tìm cây khung nhỏ nhất (MST – Minimum Spanning Tree) của đồ thị $M + 1$ cạnh bằng thuật toán Kruskal.

Độ phức tạp thời gian: $O(Q * M * log(N + M))$.

Subtask 3 (75%): $N \leq 4 * 10^3, M \leq 4 * 10^5, Q \leq 3000$.

Để cải tiến thuật toán trên, ta cần hiểu bản chất của thuật toán Kruskal. Trước hết, ta sẽ tìm cây khung nhỏ nhất T của đồ thị M cạnh. Nhận xét rằng, khi bổ sung một cạnh có trọng số 0 giữa hai đỉnh A và B, cạnh này chắc chắn sẽ nằm trong cây khung nhỏ nhất. Khi đó, đường đi giữa A và B trong T và cạnh (A, B) sẽ tạo thành một chu trình. Do thuật toán Kruskal xét các cạnh theo trọng số từ nhỏ đến lớn, cạnh có trọng số lớn nhất trên đường đi giữa A và B sẽ bị bỏ ra khỏi cây khung nhỏ nhất.



Do đó, nếu ta gọi W là tổng trọng số của các cạnh trong T, $maxW(u, v)$ là cạnh có trọng số lớn nhất trên đường đi giữa u và v trên T. Khi đó, với truy vấn i, đáp án sẽ là $W - maxW(A_i, B_i)$.

Độ phức tạp thời gian: $O(Q * N + M * \log(N + M))$.

Subtask 4 (100%): $N \leq 4 * 10^3$, $M \leq 4 * 10^5$, $Q \leq 2 * 10^5$.

Cải tiến subtask 3, ta sẽ tính trước $maxW(u, v)$ với mọi cặp (u, v) bằng cách DFS từ từng đỉnh trong $O(N^2)$, vì vậy với mỗi truy vấn ta có thể trả lời được trong $O(1)$.

Độ phức tạp thời gian: $O(M * \log(N + M) + N^2 + Q)$.