계단 만들기 풀이

키-파

November 2022

1 Small

문제를 다음과 같이 바꿀 수 있습니다.

Problem. *정수열 $\{a_i\}$가 주어졌을 때, 다음 두 조건*

- $|b_i - b_{i+1}| = 1 \quad \forall i$
- $\sum_i b_i = \sum_i a_i$

을 만족하는 정수열 $\{b_i\}$를 잡아

$$\frac{1}{2}\sum_i |a_i - b_i|$$

의 값을 최소화하라.

Proof. 원래 문제의 정답을 A라 하고, 이 문제의 정답을 B라 합시다. 또한 원래 문제의 최종 상태를 $\{c_i\}$라 합시다. $\{c_i\}$가 $\{b_i\}$에 대응하는 두 조건을 만족하는 것은 쉽게 보일 수 있습니다.

- 원래 문제에서는 A개가 최솟값이므로, 정확히 A개의 블록을 판 다음 다른 곳에 정확히 A번 놓는 것이 답이 됩니다. 이 말은

  $$\sum_{a_i > c_i} (a_i - c_i) = A$$

  이고

  $$\sum_{a_i < c_i} (c_i - a_i) = A$$

  라는 말입니다. 둘을 절댓값 기호로 합치면

  $$\sum_i |a_i - c_i| = 2A$$

  가 되고, 그러한 것 중 합을 최소화하는 것이 $\{b_i\}$이므로 $A \geq B$입니다.

- 이 문제에서는

  $$\sum_i |a_i - b_i| = 2B$$

를 만족합니다. 그런데 두 번째 조건에서

$$\begin{aligned}
\sum_i (a_i - b_i) &= 0 \\
\sum_{a_i > b_i} (a_i - b_i) + \sum_{a_i < b_i} (a_i - b_i) &= 0 \\
\sum_{a_i > b_i} (a_i - b_i) - \sum_{a_i < b_i} (b_i - a_i) &= 0 \\
\sum_{a_i > b_i} |a_i - b_i| &= \sum_{a_i < b_i} |a_i - b_i|
\end{aligned}$$

이므로, 두 값은 B로 같습니다. 이 말은 정확히 B개의 블록을 판 다음 다른 곳에 정확히 B번 놓는 답이 존재한다는 말이 됩니다. 이러한 것 중 횟수를 최소화하는 것이 A이므로, $A \le B$입니다.

따라서 $A = B$입니다. $\square$

이제 변형된 문제를 풀고, 절반으로 해 주어야 한다는 점만 까먹지 않으면서 $\sum_i |a_i - b_i|$를 최소화하는 데에 집중합시다.

Lemma 1. *정답이 되는* $\{b_i\}$ *중,* $\min_i a_i \le b_i \le \max_i a_i \quad \forall i$*인 것이 존재한다.*

Proof. $n = 1$인 경우는 자명하게 $\min_i a_i = \max_i a_i = a_1 = b_1$인 경우만이 답이 됩니다. $n \ge 2$인 경우는 귀류법을 통해 증명합니다. 최솟값 증명이 비슷하므로, 최대인 쪽만 증명합니다.

모든 b_i에 대해 $B_u := \max_i b_i > \max_i a_i$이고, $\{b_i\}$ 중 값이 B_u인 것이 정확히 c개 있는 경우가 존재하며 이것이 최소 개수라고(즉, c개 미만으로 존재하는 경우는 없다고) 합시다. 이 위치를 (아무거나 하나) k라 합시다. 또한 $b_j = \min_i b_i$인 $j \neq k$를 (아무거나 하나) 잡읍시다. $b_j < \max_i a_i$일 수밖에 없는데, 만일 $b_j = \min_i b_i \ge \max_i a_i$라면

$$\sum_i b_i = B_u + \sum_{i \neq k} b_i > \max_i a_i + (n-1) \max_i a_i \ge \sum_i a_i$$

이기 때문입니다.

이제 다음 수열을 생각합니다.

$$b'_i = \begin{cases} b_k - 1 & i = k \\ b_j + 1 & i = j \\ b_i & \text{otherwise} \end{cases}$$

$\{b'_i\}$는

- $\{b_i\}$가 만족해야 하는 조건 중 첫 번째 조건을 만족합니다. 나머지에서는 값이 같으므로, 확인해 주어야 하는 것은 k 주변과 j 주변뿐입니다.
  - k의 정의에 의해 $b_{k-1} \le b_k$이므로, $0 \le b_k - b_{k-1} = |b_k - b_{k-1}| \le 1$입니다. 이 조건에 의해

    $$b_j < \max_i a_i \le b_k - 1 \le b_{k-1}$$

    이기 때문에 $j = k - 1$일 수 없습니다. 따라서

    $$\left|b'_k - b'_{k-1}\right| = |b_k - 1 - b_{k-1}| = |(b_k - b_{k-1}) - 1| \le 1$$

    입니다 (빨간색 부분의 값이 0 혹은 1이므로).
  - 위 증명의 $(k-1)$을 $(k+1)$로 바꾸어 읽으면 k의 오른쪽도 증명됩니다.

- j의 정의에 의해 $b_{j\pm1} \geq b_j$이므로, $0 \leq b_{j\pm1} - b_j = |b_j - b_{j\pm1}| \leq 1$입니다. 또한 위 k의 경우 증명에 의해 $k \neq j \pm 1$입니다. 따라서

$$\left|b'_j - b'_{j\pm1}\right| = |b_j + 1 - b_{j\pm1}| = |(b_{j\pm1} - b_j) - 1| \leq 1$$

  입니다 (파란색 부분의 값이 0 혹은 1이므로).

- $k \neq j$이므로, 두 번째 조건은 자명하게 만족합니다.
- $|a_i - b'_i|$를 보면 k에서 1 작아지고, j에서 1 커지거나 작아지므로

$$\sum_i |a_i - b'_i| \leq \sum_i |a_i - b_i|$$

  를 만족합니다.

따라서 원래의 수열의 조건을 만족하고, 최소화하고 싶은 것이 원래의 수열 이하이면서, $b'_i = B_u$인 것이 $(c-1)$개인 수열을 찾았습니다. 이는 가정에 모순입니다. □

이 사실들을 이용해 Small을 해결할 수 있습니다. 다음과 같은 DP 식을 생각합니다.

$$D(i,t,S) = [b_i = t\text{이고 } \sum_{j\leq i} b_j = S\text{인 것들 중, } \sum_{j\leq i} |a_j - b_j|\text{의 최솟값}].$$

$1 \leq i \leq n$이고 $\min_i a_i \leq t \leq \max_i a_i$이며 $0 \leq S \leq \sum_i a_i =: \mathcal{S}$인 것에 대해서 계산해 주면, 답은

$$\frac{1}{2} \min_t D(n,t,\mathcal{S})$$

가 됩니다.

점화식은, $b_i = t$를 쓰기로 결정한 경우 직전 합은

$$\sum_{j\leq i-1} b_j = \sum_{j\leq i} b_j - b_i = S - t$$

이며, 앞선 $b_{i-1} = t-1$, $b_{i-1} = t$ 혹은 $b_{i-1} = t+1$이어야 하므로

$$\begin{aligned} D(i,t,S) = \min(&D(i-1,t-1,S-t), \\ &D(i-1,t,S-t), \\ &D(i-1,t+1,S-t) \\ &) + |a_i - t| \end{aligned}$$

가 됩니다. 초기값은 $D(1,t,t) = |a_1 - t| \quad \forall t$ 등으로 둘 수 있습니다.

시간 복잡도는 i가 n개, t가 $(\max_i a_i - \min_i a_i)$개, S가 $\sum_i a_i$개이므로

$$\mathcal{O}\left(n \cdot (\max_i a_i - \min_i a_i) \cdot \sum_i a_i\right)$$

입니다.

2 Large

문제 상황을 조금 더 관찰하기 위해 j를 고정하고 전체 수열의 합을 추정해 보도록 합시다. $b_j = v$ (for notational convenience). $c'_i = |b_i - v|$로 두고, 이 수열을 **정렬한** 수열을 $\{c_i\}$라 합니다.

Lemma 2. $c_{i+1} - c_i \le 1 \quad \forall i.$

Proof. 귀류법을 이용합니다. 만일 $c_{k+1} - c_k \ge 2$인 k가 존재한다면, $c'_i = c_k + 1$인 i는 존재하지 않습니다. $\{c_i\}$는 $\{c'_i\}$가 정렬된 수열이므로, 그런 i가 존재했다면 c_k와 $c_{k+1} \ge c_k + 2$가 붙어있을 수 없기 때문입니다.

따라서 모든 $1 \le i \le n$를 다음과 같이 두 종류로 나눌 수 있습니다.

(a) $c'_i \le c_k$인 i. c_k가 있기 때문에 반드시 하나는 존재합니다.

(b) $c'_i \ge c_k + 2$인 i. $c_{k+1} \ge c_k + 2$가 있기 때문에 반드시 하나는 존재합니다.

두 종류 모두 존재해야 하기 때문에, **어떻게 배치해도 어딘가에서 종류가 한 번 바뀌어야 합니다.** 즉 [l이 (a)이고 $(l+1)$이 (b)]이거나, [$(l+1)$이 (a)이고 l이 (b)]인 l이 존재합니다.

만일 l이 (a)이고 $(l+1)$이 (b)라면,

$$\begin{aligned} c'_l &= |b_l - v| \le c_k \\ c'_{l+1} &= |b_{l+1} - v| \ge c_{k+1} \ge c_k + 2 \end{aligned}$$

입니다. 이 말은 $b_l = v \pm c'_l$이고 $b_{l+1} = v \pm c'_{l+1}$라는 말입니다. 따라서

$$\begin{aligned} |b_l - b_{l+1}| &= \left|c'_l \pm c'_{l+1}\right| {\color{red}\le 1} \\ \text{whereas} \quad c'_l + c'_{l+1} &\ge c'_l + (c_k + 2) {\color{blue}\ge 2} \\ \left|c'_l - c'_{l+1}\right| &\ge c_{k+1} - c_k \\ &\ge (c_k + 2) - c_k {\color{blue}\ge 2} \end{aligned}$$

이므로 모순입니다. $(l+1)$이 (a)이고 l이 (b)인 경우에도 증명은 비슷합니다. □

Lemma 3. $c_i \le i - 1 \quad \forall i.$

Proof. $c'_j = |b_j - v| = 0$이고 $c'_i \ge 0 \quad \forall i$이므로, 크기 순서대로 정렬한 $\{c_i\}$에서는 $c_1 = 0$입니다. 따라서

$$c_i = c_i - c_1 = \sum_{k=1}^{i-1} (c_{k+1} - c_k) \le \sum_{k=1}^{i-1} 1 = i - 1$$

입니다. □

$\{c_i\}$는 $\{c'_i\}$를 정렬한 것뿐이므로 합이 같습니다. 따라서 삼각부등식을 이용하면

$$\begin{aligned} \left|\sum_i b_i - nv\right| &= \left|\sum_i (b_i - v)\right| \\ &\le \sum_i |b_i - v| = \sum_i c'_i \\ &= \sum_i c_i \le \sum_{i=1}^{n} (i - 1) \\ &= \frac{n(n-1)}{2} \end{aligned}$$

입니다. 그런데 문제 조건에 의해 $\sum_i b_i = \sum_i a_i =: \mathcal{S}$입니다...!

$$nv - \frac{n(n-1)}{2} \le \mathcal{S} = \sum_i b_i \le nv + \frac{n(n-1)}{2}.$$

양 부등식을 쪼개어 v에 대해 풀면,

$$v \le \frac{2\mathcal{S}+n^2-n}{2n}$$
$$v \ge \frac{2\mathcal{S}-n^2+n}{2n}$$

입니다. v가 정수여야 하므로

$$\mathcal{B}_l := \left\lceil \frac{2\mathcal{S}-n^2+n}{2n} \right\rceil \le v \le \left\lfloor \frac{2\mathcal{S}+n^2-n}{2n} \right\rfloor =: \mathcal{B}_u$$

라 합시다. 이제 $b_j = v$였으므로 j를 그만 고정하면, 다음과 같은 Lemma를 얻습니다.

Lemma 4. *$\mathcal{B}_l$과 $\mathcal{B}_u$의 위 정의를 따라가면,* ***가능한 모든*** *$\{b_i\}$**에 대해** $\mathcal{B}_l \le b_i \le \mathcal{B}_u \quad \forall i$.*

이제 $\mathcal{B}_u - \mathcal{B}_l \le n-1$임에 주목합시다. 이 말은 **합에 제한이 없어도** 이전 DP 식에서 가능한 t의 값은 $\mathcal{B}_l$과 $\mathcal{B}_u$ 사이로 최대 n개라는 말이 됩니다. $\mathcal{S}$는 굉장히 커질 수 있으나, 모든 i에 대해

$$\sum_{j \le i} (b_j - \mathcal{B}_l) \le i(n-1)$$

이므로 DP 식으로 구해야 하는 것의 개수는 $\mathcal{O}(n^2)$입니다. DP 정의를 다음과 같이 조정합시다.

$$D(i,t,S) = [b_i = t + \mathcal{B}_l\text{이고 } \sum_{j \le i} (b_j - \mathcal{B}_l) = S\text{인 것들 중, } \sum_{j \le i} |a_j - b_j|\text{의 최솟값}].$$

$1 \le i \le n$, $0 \le t < n$이며 $0 \le S \le \mathcal{S} - n\mathcal{B}_l$인 모든 쌍에 대해 계산해 주면, 답은

$$\frac{1}{2} \min_{0 \le t < n} D\left(n, t, \mathcal{S} - n\mathcal{B}_l\right)$$

이 됩니다.

잘 생각해 보면, 점화식은 차를 구할 때 t 대신 $(t+\mathcal{B}_l)$을 사용해 주어야 한다는 점만 제외하면 이전과 똑같이 계산해 주면 됩니다.

$$\begin{aligned} D(i,t,S) = \min(&D(i-1,t-1,S-t), \\ &D(i-1,t,S-t), \\ &D(i-1,t+1,S-t) \\ &) + |a_i - (t+\mathcal{B}_l)|\,. \end{aligned}$$

초기값도 약간 변형하여 $D(1,t,t) = |a_1 - (t+\mathcal{B}_l)| \quad \forall t$ 등으로 둘 수 있습니다.

시간 복잡도는 i가 n개, t가 최대 n개, S가 최대 n^2개이므로 $\mathcal{O}(n^4)$입니다.

3 메모리

그러나 이 값을 직접 계산할 경우 메모리가 잘못될 우려가 있습니다. $D(i,*,*)$를 계산할 때 $D(i-1,*,*)$에만 의존한다는 점을 이용하여 메모리를 $\mathcal{O}(n^3)$으로 줄이는 방법을 이용해 정답을 받을 수 있습니다.

또한 다음과 같은 방법을 이용할 수도 있습니다. 만일 $a_i \le \mathcal{B}_l$이면 D는 다음과 같이 구해도 됩니다.

$$D(i,t,S) = \min(\cdots) + (t+\mathcal{B}_l) - a_i = \min(\cdots) + |(t+\mathcal{B}_l) - \mathcal{B}_l| + (\mathcal{B}_l - a_i).$$

그리고 만일 $a_i \geq \mathcal{B}_u$이면, 마찬가지로 D는 다음과 같이 구해도 됩니다.

$$D(i,t,S) = \min(\cdots) + a_i - (t + \mathcal{B}_l) = \min(\cdots) + |(t + \mathcal{B}_l) - \mathcal{B}_u| + (a_i - \mathcal{B}_u).$$

빨간색은 모든 $D(i,*,*)$들에 더해지므로, 다음 D의 min 계산에 전혀 영향을 주지 않습니다. 따라서 빨간 항들을 제외해 버리기 위해 $a_i' = \min(\max(a_i, \mathcal{B}_l), \mathcal{B}_u)$로 정의하면 D의 값은 항상 n^2 이하로 계산됩니다.

이렇게 해서 값을 구할 때는 여전히 $\mathcal{S} := \sum_i a_i$로 정의된 합을 사용해야 합니다. $\{a_i'\}$가 아님에 주의합시다. 정답은 DP로 구한 답을 $\mathcal{D}$라 하면

$$\frac{1}{2}\left(2\mathcal{D} + \sum_{a_i \leq \mathcal{B}_l} (\mathcal{B}_l - a_i) + \sum_{a_i \geq \mathcal{B}_u} (a_i - \mathcal{B}_u)\right)$$

가 됩니다.

이렇게 하면 `int`나 `unsigned short`를 사용할 수 있고, 메모리 사용량을 줄여 정답을 받을 수 있습니다.