



HUST

ĐẠI HỌC BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

ONE LOVE. ONE FUTURE.



ĐẠI HỌC
BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY
OF SCIENCE AND TECHNOLOGY

Applied Algorithm Lab

Range minimum query

ONE LOVE. ONE FUTURE.

- Calculate the sum of the minimal numbers in some intervals of a given sequence
- Given a sequence of n integers a_0, \dots, a_{n-1} .

Denote $\text{rmq}(i, j)$: the minimum of a_i, a_{i+1}, \dots, a_j .

Given m pairs $(i_1, j_1), \dots, (i_m, j_m)$

Compute the sum: $Q = \text{rmq}(i_1, j_1) + \dots + \text{rmq}(i_m, j_m)$

- Input: the sequence and the intervals
- Output: the sum

- Example

stdin	stdout
16 2 4 6 1 6 8 7 3 3 5 8 9 1 2 6 4 4 1 5 0 9 1 15 6 10	6 Explain: $1+1+1+3 = 6$

- Idea to solve #1:
 - for each pair of (i, j) , find the minimum number in the interval
 - sum over all pair
- > easy, but too many comparisons, high complexity as the number of pairs goes higher

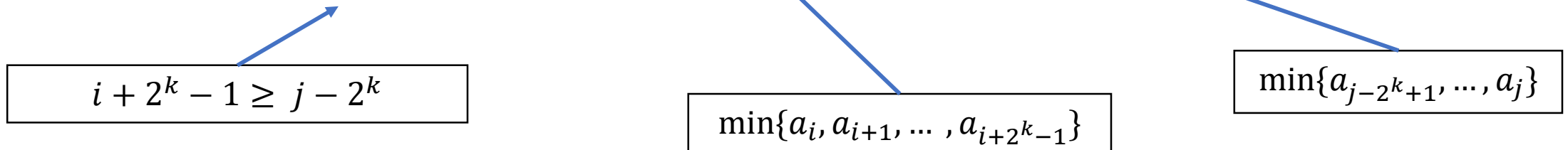
- Idea to solve #2: pre-processing and store some minimum for less-complicated comparisons for answers (faster lookups)
 - Build a table $M[N][\log_2 N]$ that:

$M[i][t]$ is the index of $\min \{a_i, a_{i+1}, \dots, a_{i+2^t-1}\}$

Then we can answer faster:

$$rmq(i, j) = \min \left(a_{M[i][k]}, a_{M[j-2^k+1][k]} \right)$$

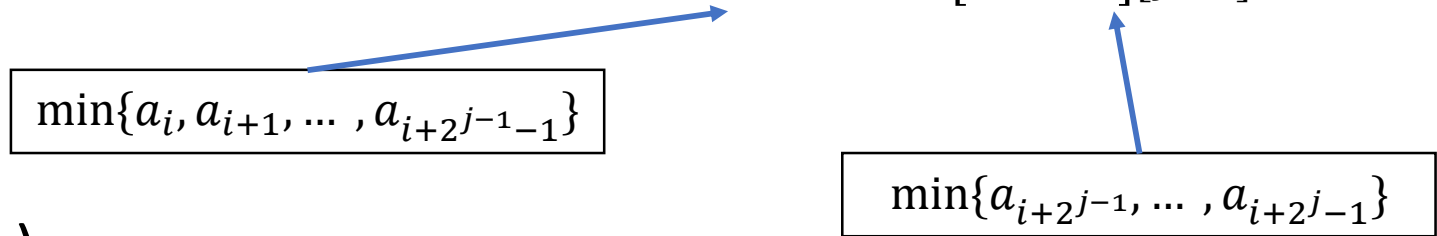
with $k = \lfloor \log_2(j - i + 1) \rfloor$



- Idea to solve #2: after determine M, it only takes $O(1)$ for each rmq query, compare to $O(n)$ as in idea #1
- To build M: use recursion
 - Base case: $M[i][0] = i$, storing the index of each element.
 - Recursive case: Uses previous results to compute larger ranges:

$M[i][j]$: the index of $\min(a_{M[i][j-1]}, a_{M[i+2^{j-1}][j-1]})$

$\min\{a_i, a_{i+1}, \dots, a_{i+2^{j-1}-1}\}$



$\min\{a_{i+2^{j-1}}, \dots, a_{i+2^j-1}\}$

- Complexity: $O(n \log n)$



HUST

THANK YOU !