

B: Merge Sequences 解説

2023/05/03 HUPC

Today03

解法

1. 二分探索
2. データ構造

二分探索解

Step 1.

与えられた数列を昇順にソートする。

二分探索解

Step 2.

K番目の値を求める。

次の二分探索によって求まる。

「 $A[i] + B[j] > X$ 」となる (i, j) がK個未満か

具体的には、

各 $A[i]$ について、 $X - A[i] > B[j]$ となる j の個数を

B に対する二分探索で求めて、合計したものがK個未満であるかどうか

この操作における計算量は $O(N \log(\max(A) + \max(B)) \log(M))$

二分探索解

Step 3.

解を構築する。（前Stepで求めたK番目の値を X とする。）

各 $A[i]$ について $A[i] + B[j] > X$ を満たす $B[j]$ をすべて列挙し
 $A[i] + B[j]$ を配列に追加する。

この配列のサイズは K 未満となるので、足りない分を X で埋め、
降順にソートして出力すればOK。

データ構造による解

(整数値, A のインデックス, B のインデックス)

の組をデータとして持ち、整数値が最大のデータを $O(\log(N))$ で取得できるデータ構造を考える。

`priority_queue<tuple<long long, int, int>>`や

セグメント木など

データ構造による解

Step 1.

与えられた数列を降順にソートする。

データ構造による解

Step 2.

各 $A[i]$ について

priority_queueに $(A[i] + B[0], i, 0)$ を追加する。

データ構造による解

Step 3.

以下の操作をK回繰り返すことで、1番目からK番目までの値を得られる。

1. priority_queueの最大値を取得する。($value, i, j$)とする。
2. $value$ を出力する。
3. priority_queueに($A[i] + B[j + 1], i, j + 1$)を追加する。($j + 1 \leq M$ の場合)

この操作における計算量は $O(K \log(N))$