



Анализ на задача *clocks*

Ще отговорим на заявките офлайн, тоест чак след като сме ги въвели и записали в масив, ще намерим отговорът за всяка от тях.

Решение за 15 точки

За 15 точки се очакват решения, които за всеки часовник i правят a_i стъпки, и на всяка стъпка пазят колко показва часовника. В един масив можем да пазим колко пъти сме минали през всеки час. След като направим това за всички часовници, ще сме готови да отговорим на заявките.

Сложността на решението е $O(N * A)$, където $A = \max\{a_1, \dots, a_n\}$.

Решение за 30 точки

За 30 точки ограничението за издръжливостта на всеки часовник става 10^{12} . Можем да забележим, че не е нужно да минаваме през всички часове на всеки часовник, защото те се повтарят през M . Със сигурност часовник i ще мине през всеки час $\lfloor \frac{a_i}{M} \rfloor$ пъти. Нека $q_i = a_i \% M$. Часовете, които остава да преброим, можем да ги представим като един или два затворени интервала $[l, r]$. Единият интервал е $[a_i, \min(M - 1, a_i + q_i)]$ и ако $q_i > M - a_i$, другият интервал е $[0, q_i - (M - a_i) - 1]$. В една променлива sum ще запазим $\sum_{i=1}^N \lfloor \frac{a_i}{M} \rfloor$. Остава за всяка заявка t_i да преброим колко са интервалите, които покриват t_i и към това число да прибавим sum . За 30 точки е достатъчно за всяка заявка просто да обходим интервалите. Сложността е $O(Q * N)$.

Решение за 100 точки

За 100 точки, за да пресметнем колко интервала покриват всяко t_i , ще използваме техниката "метяща права" (sweepline). Интересуваме се от 3 вида точки - начала на интервал, краища на интервал и заявки. Сортираме всички точки (за всяка пазим какъв е типът ѝ) и ги обхождаме. Ще пазим в една променлива колко са незатворените интервали към момента. Когато видим начало на интервал, увеличаваме променливата с 1. Когато видим край на интервал, я намаляваме с 1. Така когато видим заявка t_i , ще знаем колко интервала съдържат t_i .

Остава да кажем една особеност на решението и това е как да сортираме равните точки. Редът, в който трябва да бъдат обходени тези точки, е следният: първо са началата на интервали, на второ място са заявките, на които да отговаряме, и последни са краищата на интервали. По този начин си гарантираме, че ще отговорим правилно на заявките.

Сложността на решението е $O((N + Q) \log(N + Q))$ от сортировката.