

Лечебный смузи

Автор задачи и разработчик: Егор Юлин

Отметим некоторые факты, после чего предложим алгоритм построения массива, в котором будет максимальный ответ.

Если в массиве встречается число a_i , при этом оно уже было раньше, то ничего не изменится. Из-за этого все повторения чисел можно расположить в конце в любом порядке.

Обозначим $b_i = \text{mex}(a_1, \dots, a_i)$. Заметим, что массив b не убывает. Кроме того, все повторения чисел мы переместили в конец, тогда рассмотрим массив a , в котором все числа различны. Чтобы получить максимальный $\text{mex}(b)$, мы хотим получить массив b вида $(0, 1, 2, 3, \dots)$.

Докажем, что если отсортировать массив a , после чего расположить его элементы как $(a_n, a_1, a_2, a_3, \dots)$, то массив b получится оптимальным.

Действительно,

- первое число должно быть не равно 0, т.к. иначе $b_1 = 1$ и итоговый mex будет равен 0;
- чтобы на второй позиции в массиве b получить 1, нужно расположить где-то 0, но это не первая позиция, поэтому 0 должен стоять на второй позиции;
- и так далее;
- среди всех элементов на первой позиции должен стоять максимальный, потому что число на первой позиции в массиве a никогда не появится в массиве b .

Тогда можно отсортировать a , все неуникальные числа перенести в конец, f на первую позицию поставить максимум. После этого останется посчитать mex на префиксах (это делается с помощью `std::map`), после этого посчитать итоговый mex . Время работы: $\mathcal{O}(n \log n)$.