

## Анализ на задача Мандарини

Наивен подход за решаване на задачата е да се симулира всяко движение. По-точно във всеки ход да намерим две най-малки числа, ако е ред на първия продавач, или две най-големи числа, ако е ред на втория продавач. След това правим съответната промяна и симулираме тази процедура, докато играта приключи. Тъй като увеличението на броя на ходовете е квадратично, като се има предвид броя мандарините и че ние необходимо време  $O(n)$ , за да намерим най-малките/най-големите числа, общата времева сложност на този подход е  $O(n^3)$ , което е достатъчно, за да получите 50% от общия брой точки.

Като използваме по-интелигентно съхранение на данните, можем да спестим време, необходимо за намиране на най-малките/най-големите числа. За тази цел ще използваме структурата `deque`, която ни позволява да манипулираме данни от горната или долната част на структурата със сложност  $O(1)$ . Всеки елемент от структурата ще бъде подредена двойка (брой кашони, количество мандарини), която обозначава броя на кашоните с еднакъв брой мандарини с тях и елементите ще бъдат сортирани във възходящ ред според количеството мандарини (отгоре надолу). Търсенето на екстремни стойности сега се свежда до премахване на елементите от горната/долната част на конструкцията. Освен това симулацията на играта се свежда до актуализиране на първите две стойности от горната или долната част на структурата (в зависимост от това кой е на ред). Тази и подобни реализации, които имат времева сложност  $O(n^2)$ , са достатъчни, за да получат 80% от точките за тази задача.

И накрая, нека забележим, че при множество най-голям и най-малък брой мандарини, някои ходове ще бъдат повторени.

Да илюстрираме това с пример. Нека броят на мандарините в кашоните е  $\{1,1,1,2,3,4,4,4\}$ . Първите няколко хода (ако първият продавач започне) ще бъдат:

- увеличете 1 на 2
- намалете 4 на 3
- увеличете 1 на 2
- намалете 4 на 3
- увеличете 1 на 2

Едва сега намалихме броя на кашоните с различно количество мандарини с 1. Състоянието на кашоните след тези ходове е  $\{2,2,2,2,3,3,3,4\}$ .

Използвайки идеите от предишните параграфи, можем лесно да анализираме броя на кашоните с най-малко/най-много мандарини и да заключим какво ще бъде състоянието, след като броят на различните количества се намали/увеличи с 1. Следователно, при всеки ход на алгоритъма със сигурност ще намалим броя на елементите в структурата `deque` с 1.

Тази и подобни реализации, които имат времева сложност  $O(n)$ , са достатъчни, за да получат 100 точки.