

## ЗАДАЧА D1. СТЬЛБА

За пълно решение на задачата използваме вариант на техниката с плъзгане на прозорец. За да получим стълба от каменните блокове в обичайния смисъл на думата, блоковете трябва да се подредят в нарастващ ред на височините. Затова трябва първо да сортираме редицата от височини със стадартната функция за сортиране със сложност  $O(N \log N)$ .

### Решение със сложност $O(N \log N)$

Тъй като единствената характеристика на прозореца, която се търси е броят на елементите в него, не е нужно да отчитаме друго за прозореца, освен търсения брой. Затова започваме с прозорец с един елемент и докато височината на следващия в сортирания ред блок е **различна от височината на последния избран** и не по-голяма от  $K$ , отваряме прозореца с един елемент на дясно и увеличаваме броя на блоковете в разглеждания прозорец. Когато следващият блок е с височина по-голяма от  $K$ , сравняваме дължината на намерения прозорец с дължината на най-дългия до момента и ако тя е по-голяма от запомнената – заменяме запомнената с новата по-голяма. Елементът с по-голяма височина, който не е влязал в текущия прозорец става първи елемент на новия и продължаваме плъзгането.

Както винаги при техниката с плъзгане на прозорец, не трябва да забравяме, че най-добрият прозорец може да завършва с последния елемент на редицата. Затова след завършване на основния цикъл трябва непременно да проверим неговата дължина и да обновим търсения максимум, ако се налага. Сложността на това плъзгане на прозореца е  $O(N)$  и така сложността на решението е равна на сложността на сортирането  $O(N \log N)$ . Това решение може да се види във файла `author.cpp`.

### Решение със сложност $O(N^2)$

Нашата задача може да се разглежда като частен случай на задачата за намиране на най-дълга растяща подредица. В [https://cp-algorithms.com/sequences/longest\\_increasing\\_subsequence.html](https://cp-algorithms.com/sequences/longest_increasing_subsequence.html) могат да се видят решения със сложност  $O(N^2)$  и  $O(N \log N)$ . В `stairs_70p.cpp` е адаптиран алгоритъмът със сложност  $O(N^2)$ . Това решение не може да реши големите тестове в предвиденото време.

Ако с едно минаване по масива се отстранят равните елементи, достигаме до решение `stairs_100p.cpp`, което е със сложност  $O(N^2)$ , но може да получи 100 точки, защото вместо с масив от 1 милион елемента, вече работи с масив от най-много 10 хиляди елемента.

*Красимир Манев, Стоян Капралов*