Ако $x\_{i}\geq c\*\frac{x\_{1}+…+x\_{k}}{k}$ е спазено за най-малкото $x\_{i}$, то е спазено за всяко $x\_{i}$, затова всеки път трябва да увеличаваме най-малкото $x\_{i}$.

**Решение за 5 точки**

Директна симулация.

Сложност: $O(q\*n^{2}\*x\_{i})$

**Решение за 15 точки**

Искаме да намерим най-малкото $v$, за което $v\geq c\*\frac{max(x\_{1}, v)+…+max(x\_{k}, v)}{k}$, което след сортиране на числата може да се намери линейно.

Сложност: $O(q\*n\*log\_{2}(n)))$

**Решение за 40 точки**

Можем да правим $binary search$ по$ v$ с $merge sort tree$ или $persistent segment tree$ (което изисква твърде много памет за следващите събтаскове), а може и с $sqrt decomposition$ да поддържаме кои числа са в интервала и по тях да правим $binary search$ по$ v$, като структурите трябва да могат да отговорят на въпроса “По дадени числа $b\_{l}, …, b\_{r}$, колко е броят и каква е сумата на по-малките от $x$?”

Сложност: $O(q\*log\_{2}\left(n\right)^{3})/O\left(q\*log\_{2}\left(n\right)^{2}\right) с много памет/O(n\*\sqrt{q}\*log\_{2}(n))$

**Решение за 65 точки**

Оптимизирани решения на предния събтаск или неоптимизирани на следващия.

**Решение за 100 точки**

$parallel binary search$ като въпросът е по $i$ на сортираните $a$:

“Ако считаме, че $a\_{i}\leq v\leq a\_{i+1}$ и третираме $a\_{j}\leq a\_{i}$ като $v$, то ако стойността $v'$, която е уникалното решение на $v=c\*\frac{max(x\_{1}, v)+…+max(x\_{k}, v)}{k}$ (или $0$, ако $c=1 и v\geq x\_{1}, …, x\_{k}$), e $a\_{i}\leq v^{'}\leq a\_{i+1}$, сме намерили желаното $v$, ако $v^{'}>a\_{i+1}$, то $v>a\_{i+1}$, ако $v'<a\_{i}$, то $v<a\_{i}$”, което може да поддържа с $fenwick tree$

Сложност: $O\left((n+q)\*log\_{2}\left(n\right)^{2}\right) )$

 Автор: Мартин Копчев