

АНАЛИЗ

Подзадача 1

Решението на тази подзадача е просто да сравним всеки две монети с номера, различаващи се точно с 1. Тъй като по условие има поне една тежка и поне една лека монета, няма как стойностите, получени от функцията `weigh` да бъдат само 0. Когато получим отговор, различен от 0, директно намираме една лека и една тежка монета и според резултатите от останалите сравнения определяме вида на всяка една монета. Общият брой извиквания на функцията е $O(2^K)$.

Подзадача 2

В тази подзадача знаем, че сред множеството от монети има $N - 1$ еднакви по тежест монети и една различна. Целта ни ще бъде да намерим различната, без значение дали тя е тежка или лека. Ако $K = 1$, това е тривиално, затова нека се съсредоточим върху случая, когато $K \geq 2$. Нека начално разделим множеството на две равни по големина части, които ще означим L и R . Очевидно резултатът от сравнението на L с R ще бъде ± 1 , но той не ни дава пълна информация, понеже има две възможни ситуации. В случай че $L > R$ (множеството L е по-тежко от R) тези ситуации са:

- L се състои от $\frac{N}{2}$ тежки монети, а R от $\frac{N}{2} - 1$ тежки и една лека;
- L се състои от $\frac{N}{2} - 1$ леки и една тежка монета, а R се състои от $\frac{N}{2}$ леки монети.

Забелязваме, че ако изберем едно от двете подмножества – например L , и го разделим на две части, които ще означим L_1 и L_2 , то резултатът от претеглянето на L_1 и L_2 определя в кой от двата случая сме. Ако резултатът е 0, то тогава L се състои от монети с еднаква тежест и обратното – ако резултатът не е 0, тогава в L има поне една лека и поне една тежка монета.

Оттук нататък можем да приложим стандартно двоично търсене, като на всяка итерация разделяме оставащото множество на две равни по големина части, сравняваме ги и продължаваме търсенето в частта, в която има поне една лека и поне една тежка монета, докато не останат точно две монети. Общият брой извиквания на функцията `weigh` е $O(K)$.

Подзадача 3

Решението на тази подзадача също се основава на двоично търсене, но не в най-стандартната му форма. От ограничението, че сред всички N монети има най-много K , които са с тежест, различна от преобладаващата, следва, че броят на „блоковете“ от последователни монети с еднаква тежест е най-много $2K + 1$. Ще използваме двоично търсене, за да намерим дължината на всеки блок, започвайки от неговото начало. Нека от монета с номер p започва блок от последователни монети с еднаква тежест. Ако $p = N - 1$, то очевидно този блок има дължина 1. Иначе сравняваме монетите с номера p и $p + 1$. Ако резултатът от сравнението е ± 1 , то дължината на блока отново е 1, но ако получим 0, то дължината е поне 2. Продължаваме по същия начин, като ако знаем, че от монетата p започва блок с дължина поне 2^k , сравняваме множествата от монетите с номера в интервалите $[p, p + 2^k - 1]$ и $[p + 2^k, p + 2^{k+1} - 1]$. Ако получим равенство, вече ще знаем, че от монетата p започва блок с дължина поне 2^{k+1} . Ако сравнението е невъзможно (понеже $p + 2^{k+1} > N$) или получим неравенство, то дължината на блока е в интервала $[2^k, 2^{k+1} - 1]$.

Чрез още едно двоично търсене можем да намерим номера q на последната монета от блока. Начално знаем, че q е в интервала $[l, r]$ (получен от първото двоично търсене) със среда $m = \frac{l+r}{2}$. Можем да сравним монетите с номера в интервалите $[l, m]$ и $[2l - m - 1, l - 1]$, като за тези от втория интервал сме сигурни, че имат еднаква тежест. Ако резултатът от сравнението е равенство, то q ще бъде в интервала $[m, r]$, а в противен случай в интервала $[l, m - 1]$. Общият брой сравнения, използван от това решение, е $O(K \log \frac{N}{K})$.

Подзадача 5

Авторов подход

Нека създадем процедура, която разделя дадено множество от $N = 2^K$ монети на две подмножества от по $\frac{N}{2}$ монети, така че броят на тежките монети в двете подмножества се различава най-много с 1.

Прилагаме произволно разделяне, за да получим две начални подмножества L и R .

Във всяко от тях ще имаме фиксирани и нефиксирани монети, като само нефиксираните ще могат да бъдат премествани в другото подмножество. В началото всички монети са нефиксирани.

Уверяваме се, че $L > R$ – ако не е така, разменяме двете подмножества.

1. Вземаме половината от нефиксираните монети от L и половината от нефиксираните монети от R и ги разменяме.
2. Ако отново $L > R$, фиксираме монетите, които току-що сме разменили.
3. Ако $L < R$, връщаме разменените монети в предишните им подмножества, фиксираме тези от нефиксираните, които не са участвали в разместването.
4. Ако $L = R$, приключваме.
5. Докато има повече от една нефиксирана монета във всяко от двете подмножества, се връщаме на стъпка 1.

Използвайки тази процедура, преброяването е доста лесно. Ако двете подмножества са еднакво тежки, тогава броят на тежките монети в цялото множество е равен на удвоения брой на тежките монети в едно от двете подмножества. В противен случай броят на тежките монети в цялото множество е с едно по-малък от удвоения брой на тежките монети в по-тежкото подмножество. Разбира се, за да е по-тежко от другото, това множество трябва да съдържа поне една тежка монета. Тази информация ни е необходима, ако в процеса на рекурсивното изчисляване получим подмножество, състоящо се само от един вид монети.

Решението използва $O(K^2)$ сравнения и има времева сложност $O(N \log N)$.

Доказателство на коректността на описаната процедура:

Нека имаме две непресичащи се подмножества L и R , такива че $L > R$.

С d ще означим разликата на броя фиксирани тежки монети в L и R (начално $d = 0$).

С x ще означим разликата на броя нефиксирани тежки монети в L и R .

Ще докажем следния инвариант: $|d| < x$

Основно свойство: инвариантът е в сила в началото, освен ако $x = 0$, но тогава процедурата приключва и връща верен резултат

Индуктивно свойство:

Нека $x = y + z$, където y и z са разликите на неизбраните и избраните за разместване от нефиксираните монети в двете подмножества

I) $y + z - d > 0$ (от инварианта)

II) $d + y + z > 0$ (от $L > R$)

1 сл. III) $d + y - z > 0$ (от сравнението)

$$d_{new} = d - z$$

$$x_{new} = y$$

$$\text{От III)} \Rightarrow d - z > -y \text{ и от I)} \Rightarrow |d| - z < y \Rightarrow |d - z| < y \Rightarrow |d_{new}| < x_{new}$$

2 сл. III) $d + y - z < 0$ (от сравнението)

$$d_{new} = d + y$$

$$x_{new} = z$$

$$\text{Om III)} \Rightarrow d + y < z \text{ u om II)} \Rightarrow d + y > -z \Rightarrow |d + y| < z \Rightarrow |d_{new}| < x_{new}$$

Крайност: Броят на нефиксираните монети намалява наполовина на всяка итерация и алгоритъмът приключва, когато остане само една такава във всяко от подмножествата.

Коректност на постусловието: $|d| < x \leq 1$, следователно постусловието е изпълнено.

Алтернативен подход

Съществува алтернативно решение, използващо същата стратегия, но с различна процедура за разделяне на монетите в две подмножества, базирана на наблюдение за непрекъснатост на функция.

Нека отново разделим монетите на подмножества, като в едното подмножество включим всички монети с номера в интервала $[l, r]$, където $r - l + 1 = \frac{N}{2}$, а в другото подмножество включим всички останали монети. Правим наблюдението, че ако решим да променим текущото разделяне, така че вече първото подмножество да включва монетите с номера в интервала $[l + 1, r + 1]$, броят на тежките монети в него ще се промени най-много с 1. Това е така, защото при описаната промяна най-много една тежка монета се присъединява към първото подмножество и най-много една тежка монета го напуска.

Нека дефинираме функцията $f(x)$ като броя на тежките монети с номера в интервала $[x, x + \frac{N}{2} - 1]$ и нека $p = f(0)$ и $q = f(\frac{N}{2})$. Можем да изразим предходното наблюдение като: $|f(x) - f(x + 1)| \leq 1$ за $\forall x \in [0, \frac{N}{2} - 1]$. Освен това е ясно, че или $p \leq \left\lfloor \frac{H}{2} \right\rfloor \leq q$ или $q \leq \left\lfloor \frac{H}{2} \right\rfloor \leq p$, където H е броят на тежките монети. Иначе казано, броят на тежките монети в едната половина е по-малък или равен на половината от тях, а в другата половина е по-голям или равен на половината от тях. Следователно, със сигурност $f(x)$ придобива стойност $\left\lfloor \frac{H}{2} \right\rfloor$ за някакво $x \in [0, \frac{N}{2}]$. Именно стойността на това x ни трябва, за да намерим подходящо разделяне на монетите в две подмножества.

Дали стойността на $f(x)$ е по-голяма, по-малка или равна от $\frac{H}{2}$ можем да установим лесно с едно извикване на функцията `weigh`, подавайки като параметри подмножествата от монети с номера в интервалите $[x, x + \frac{N}{2} - 1]$ и $[0, x - 1] \cup [x + \frac{N}{2}, N - 1]$.

Сега ще използваме двоично търсене в интервала $[l, r]$, като в началото $l = 0$ и $r = \frac{N}{2}$. Както вече показвахме, ако $m = \min(f(l), f(r))$ и $M = \max(f(l), f(r))$ и $m \leq \left\lfloor \frac{H}{2} \right\rfloor \leq M$, то със сигурност съществува $x \in [l, r]$, за което $f(x) = \left\lfloor \frac{H}{2} \right\rfloor$. Проверяваме стойността на $f(x)$ в средата на интервала и избираме тази половина от него, в която условието $m \leq \left\lfloor \frac{H}{2} \right\rfloor \leq M$ продължава да бъде в сила. Така накрая ще намерим подходяща стойност за x , даваща ни разделяне, в което броят на тежките монети в двете подмножества се различава с най-много 1. Сложността по брой заявки и по време е сравнима с тази на авторовото решение.

Автор: Добрин Башев