Informatyczny kącik olimpijski (130): *Kwadraty liczb naturalnych*

W tym odcinku omówimy rozwiązanie zadania „Kwadraty liczb naturalnych”, które pojawiło się na finale Zawodów Indywidualnych XIII Młodzieżowej Olimpiady Informatycznej.

Kwadraty liczb naturalnych: *Piotr jest zafascynowany kwadratami liczb naturalnych, czyli liczbami:* $1, 4, 9, 16, 25, \ldots$ *Chłopiec ma przed sobą ciąg* n *liczb naturalnych* $a = (a_1, a_2, \ldots, a_n)$. *Ile jest takich par liczb naturalnych w ciągu* a, *że ich iloczyn jest kwadratem liczby naturalnej?*

Niech m oznacza wartość największej liczby w ciągu a, zaś M niech oznacza największy iloczyn pary liczb w ciągu a.

Rozwiązanie $O(n^2 \cdot \sqrt{M})$
Zacznijmy od rozwiązania, w którym obliczamy iloczyn każdej z $\frac{n(n-1)}{2}$ par elementów ciągu a. Następnie zliczamy te iloczyny, które są kwadratami liczb naturalnych. Załóżmy, że chcemy sprawdzić, czy x jest kwadratem. W tym celu przeglądamy kwadraty kolejnych liczb naturalnych: $2^2, 3^3, 4^2, \ldots$ Jeśli trafimy na x, to odpowiedź jest pozytywna. Jeśli trafimy na liczbę większą niż x, wtedy przerywamy działanie i odpowiedź jest negatywna. Tym sposobem rozważymy $O(\sqrt{x})$ kwadratów. Zatem rozwiązanie działa w czasie $O(n^2 \cdot \sqrt{M})$.

Rozwiązanie $O(n^2 \cdot \log(m))$
Przyspieszmy sprawdzanie, czy x jest kwadratem. Zaaplikujmy algorytm wyszukiwania binarnego na ciągu kwadratów: $1^2, 2^2, \ldots, x^2$. Algorytmy wykona $O(\log(x))$ operacji. Całe rozwiązanie działa w czasie $O(n^2 \cdot \log(m))$.

Szkic rozwiązania optymalnego
Zauważmy, że w rozkładzie na czynniki pierwsze kwadratu liczby naturalnej każdy czynnik pierwszy występuje parzyście wiele razy. Zatem iloczyn dwóch liczb jest kwadratem, jeśli zbiory czynników pierwszych występujących nieparzyście wiele razy w rozkładzie obu liczb są takie same. Niech więc $b = (b_1, b_2, \ldots, b_n)$, gdzie b_i oznacza iloczyn czynników pierwszych, które występują nieparzyście wiele razy w rozkładzie a_i. Innymi słowy, b_i to a_i podzielone przez swój największy dzielnik będący kwadratem. Wówczas $a_i \cdot a_j$ jest kwadratem, jeśli $b_i = b_j$. Zatem wynikiem jest liczba par takich samych elementów w ciągu b.

Zliczanie par
Załóżmy przez chwilę, że znamy ciąg b i chcemy policzyć liczbę par elementów o takich samych wartościach. W tym celu wystarczy posortować ciąg niemalejąco. Wówczas elementy o tych samych wartościach znajdą się obok siebie (będą tworzyły bloki). Następnie przeglądamy kolejne bloki elementów i zliczamy pary. W k-elementowym bloku można wybrać $\frac{k(k-1)}{2}$ par. Sortowanie realizujemy w czasie $O(n \cdot \log(n))$, podział na bloki odbywa się w czasie $O(n)$. Zatem zliczanie par tych samych elementów w ciągu b odbywa się w czasie $O(n \cdot \log(n))$.

Wyznaczenie ciągu b w $O(n \cdot \sqrt{m})$
Wartość b_i dla każdego i od 1 do n liczymy niezależnie. Początkowo niech $b_i = a_i$ dla każdego i. Następnie przeglądamy kolejne kwadraty $(2^2, 3^2, \ldots, (\lfloor\sqrt{a_i}\rfloor)^2)$ jako kandydatów na dzielniki a_i. Jeśli rozpatrywany kwadrat jest dzielnikiem b_i, to dzielimy b_i przez ten kwadrat. W ten sposób z rozkładu na czynniki pierwsze usuniemy kwadraty.

Wyznaczenie ciągu b w $O\big(n \cdot \frac{\sqrt{m}}{\log(m)}\big)$ $(*)$
Zauważmy, że nie ma potrzeby przeglądania wszystkich kwadratów. Wystarczy rozważać kwadraty liczb pierwszych $(2^2, 3^2, 5^2, \ldots)$. Pamiętajmy o tym, że kwadrat może wielokrotnie występować w rozkładzie na czynniki pierwsze. Wtedy musimy usunąć wszystkie jego wystąpienia. Przykładowo: liczbę $48 = 2 \cdot 2 \cdot 2 \cdot 2 \cdot 3$ musimy dwukrotnie podzielić przez 2^2. Każda liczba zostanie podzielona co najwyżej $O(\log(m))$ razy. Liczbę liczb pierwszych w przedziale $[1; \lfloor\sqrt{m}\rfloor]$ szacujemy jako $\frac{\sqrt{m}}{\log(m)}$. Stąd całkowita złożoność wynosi $O\big(n \cdot \frac{\sqrt{m}}{\log(m)}\big)$.

Wyznaczenie ciągu b w $O(n \cdot \log(m) + m)$ $(**)$
Na początku zliczmy wystąpienia liczb w ciągu a. Niech $Z_w = \{i \mid a_i = w\}$ (zbiór indeksów elementów ciągu a o wartości w). Następnie przeglądamy kwadraty liczb pierwszych z przedziału $[1; m]$ jako potencjalne dzielniki. Załóżmy, że rozpatrujemy kwadrat p^2. Wiemy, że jest on dzielnikiem: $1 \cdot p^2, 2 \cdot p^2, 3 \cdot p^2, \ldots, \lfloor\frac{m}{p^2}\rfloor \cdot p^2$. Dla każdej z tych wartości, korzystając z tablicy zliczającej Z, odwołujemy się bezpośrednio do elementów, które są podzielne przez p^2, i te elementy dzielimy przez p^2. Po zakończeniu otrzymane wartości tworzą ciąg b. Do każdego elementu ciągu odwołamy się co najwyżej $O(\log(m))$ razy. Liczba rozpatrywanych wielokrotności wynosi: $\frac{m}{2^2} + \frac{m}{3^2} + \frac{m}{5^2} + \ldots$, co w sumie wynosi $O(m)$. Zatem otrzymaliśmy rozwiązanie działające w czasie $O(n \cdot \log(m) + m)$.

Wyznaczenie ciągu b w $O\big(n \cdot \frac{\sqrt[4]{m}}{\log(m)} + m^{\frac{3}{4}}\big)$
Połączmy dwa poprzednie rozwiązania $(*)$ i $(**)$. Najpierw, podobnie jak w $(*)$, dla każdego elementu ciągu a rozważmy jego potencjalne dzielniki (kwadraty liczb pierwszych), nie większe niż $\sqrt{m}$. Takich dzielników jest $O\big(\frac{\sqrt[4]{m}}{\log(m)}\big)$. Następnie zaaplikujmy rozwiązanie $(**)$ z drobną zmianą. Rozpatrujmy tylko wielokrotności kwadratów liczb pierwszych większych od $\sqrt{m}$. Niestety, tak opisane rozwiązanie nadal wymaga utworzenia tablicy zliczającej rozmiaru $O(n + m)$. Do implementacji tablicy zliczającej wystarczy użyć tablicy mieszającej (tablicy z haszowaniem) i problem zostaje rozwiązany. Otrzymujemy algorytm, który działa w czasie $O(n \cdot \frac{\sqrt[4]{m}}{\log(m)} + m^{\frac{3}{4}})$. Dowód tego oszacowania pozostawiam Czytelnikowi. Jednocześnie zachęcam do pomyślenia nad lepszym oszacowaniem.

Bartosz ŁUKASIEWICZ