Тагове	На пълното решение	На подзадачите
	Метод на показалките	Двоично търсене

Анализ

Подзадача №1

В тази подзадача е тестовият пример. Тя е за обратна връзка от системата.

Подзадача №2

Разглеждат се всички подмасиви (приблително N^2 на брой) и за всеки от тях се сумира произведението на числата по двойки (двойките са приблизително дължината^2 на брой).

Постигната сложност: $O(N^4)$.

Имплементация: `First_15p.cpp`

Подзадача №3

Вместо да се обхождат всички възможни двойки в един подмасив, може да се забележи, че за i-тия боксьор, произведеният екшън е $(a_l + a_{l+1} + \cdots + a_{i-1}) \times a_i$. Сборът на числата от l до i се смята в отделна променлива, като по този начин се намира произведеният екшън в един турнир за линейно време.

Постигната сложност: $O(N^3)$.

Имплементация: `Second_30p.cpp`

Подзадача №4

Нека означаваме екшънът, произведен за масивът $[l, r]$ с $f(l, r)$. Може да се забележи, че $f(l, r + 1) = f(l, r) + (a_l + a_{l+1} + a_{l+2} + \cdots + a_r) \times a_{r+1}$, по аналогия с подобрението в горната подзадача. Заради това аналогично на горната подзадача поддържаме сбора на числата от l до r в променлива и смятаме екшънът в от един турнир за константно време.

Постигната сложност: $O(N^2)$.

Имплементация: `Third_45p.cpp`

Подзадача №5

Вместо да смятаме подмасивите с екшън ≥ 1, ще намерим подмасивите с екшън $= 0$. За това, обхождаме масива отляво-надясно, като поддържаме първият и вторият елемент ≥ 1 отляво на текущия. Решението много наподобява на задачата B2 Even от НОИ1 2022 година.

Постигната сложност: $O(N)$.

Имплементация: `Forth_20p.cpp`

Подзадача №6

Нека намерим по-добър начин за изчисление на $f(l,r)$. Това ще го направим, като съберем всяка двойка по два пъти. Тогава трябва да се намери сбора на $a_i \times (a_l + a_{l+1} + \cdots a_r - a_i)$ за всяко $l \leq i \leq r$. Тогава $2 \times f(l,r) = a_l \times (a_l + a_{l+1} + \cdots + a_r - a_l) + a_{l+1} \times (a_l + a_{l+1} + \cdots + a_r - a_{l+1}) + \cdots + a_r \times (a_l + a_{l+1} + \cdots + a_r - a_r) = a_l \times (a_l + a_{l+1} + \cdots + a_r) + a_{l+1} \times (a_l + a_{l+1} + \cdots + a_r) + \cdots + a_r \times (a_l + a_{l+1} + \cdots + a_r) - a_l^2 - a_{l+1}^2 - \cdots - a_r^2 =$

$$(a_l + a_{l+1} + \cdots + a_r) \times (a_l + a_{l+1} + \cdots + a_r) - a_l^2 - a_{l+1}^2 - \cdots - a_r^2.$$

Чрез това изразяване може да се пресмята сбора на елементите по двойки чрез префиксни суми, едната за сбор на самите елементите в редицата, дргугите за сбор на квадратите на числата в редицата. Чрез двоично търсене се намира най-дясната позиция j за всеки елемент i, за която $f(i,j) \leq k - 1$. Всички подмасиви с начало i и край $p \geq j + 1$ са с $f(i,p) \geq k$.

Постигната сложност: $O(N\ log_2\ N)$.

Имплементация: `fifth_85p.cpp`

Подзадача №7

Прилага се същата идея като в горната подзадача, само че вместо двоично търсене се използва метода на показалките.

Постигната сложност: $O(N)$.

Имплементация: `author_100p.cpp`

Автор: Борис Михов