

Xor

СЕЗОН 7 – ШЕСТИ РУНД



Както всички програмисти, Иванчо е много заинтригуван от побитовата операция xor. В опитите си да научи повече за нея, той попадна на следната задача:

Даден е масив с N елемента и цяло неотрицателно число K . Търси се броя на двойките индекси i, j ($1 \leq i < j \leq N$), такива че $A_i \oplus A_j = K$, където \oplus е побитовата операция изключващо или (xor).

От вас, както обикновено, се иска да помогнете на Иванчо като решите дадената задача.

Вход

От първия ред на файла `xor.in` се въвеждат две числа N и K – броят числа в редицата и числото K .

От втория ред се въвеждат N числа разделени с интервал – елементите на масива A .

Изход

На единствен ред в изходния файл `xor.out` отпечатайте едно число – търсеният брой двойки.

Забележка: Четенето от файл и печатането във файл става като добавите оператор за това. Можете да използвате оператор `freopen` като включите библиотека `cstdio` и добавите следните два реда в началото на `main` функцията си: `freopen("xor.in", "r", stdin);` `freopen("xor.out", "w", stdout);`

Ограничения

$$1 \leq N \leq 100\ 000$$

$$0 \leq K \leq 100\ 000$$

$$1 \leq A_i \leq 100\ 000$$

Ограничение за време: 0.5 сек

Ограничение за памет: 256 MB

Примерен тест

Вход (<code>xor.in</code>)	Изход (<code>xor.out</code>)
3 3	1
1 2 3	

Xor

СЕЗОН 7 – ШЕСТИ РУНД



6 1	2
1 4 1 5 3 2	

Пояснения

В първия пример единствената двойка е при $i = 1, j = 2$, защото $1 \oplus 2 = 3 = K$.

В втория пример двете двойки са $i = 2, j = 4$ (защото $4 \oplus 5 = 1 = K$) и $i = 5, j = 6$ (защото $2 \oplus 3 = 1 = K$). Следователно отговорът е 2.

Можете да прочетете повече за побитовата операция хог тук:

https://en.wikipedia.org/wiki/Bitwise_operation#XOR.