CHEFEXQ - Editorial

1

PROBLEM LINK:

Practice
Contest

Author: Yogendra Bhanu Singh
Tester: Mugurel Ionut Andreica
Editorialist: Kirill Gulin

PROBLEM

You have an array of N elements and 2 types of queries:

1. Given two numbers i and x, the value at index i should be updated to x.
2. Given two numbers i and k, your program should output the total number of prefixes of the array with the last index $\le i$ in which the xor of all elements is equal to k.

QUICK EXPLANATION

Replace the array by its prefix xors array. The queries became as follows: xor the suffix of the array with some value and find how many indexes less than a given one have a value equal to the some other given value. For doing this split the array into blocks of size $O(\sqrt{N})$ and perform queries over the blocks.

EXPLANATION

0-indexation is used in the editorial. Symbol $\oplus$ means bitwise xor.

Denote $p[i]$ as bitwise xor of the first i elements of the array, i.e. $p[i] = a_0 \oplus a_1 \oplus \ldots \oplus a_i$. $p[i]$ is called prefix xors sums array of the array a and can be calculated in $O(n)$ time using the fact $p[i] = p[i-1] \oplus a[i]$ for $i \ge 1$. Replace array a with it's prefix xor array p and perform queries over the array p, not a.

Suppose it's need to perform a query of the second type, i.e. find the amount of prefixes of the array a with its length no more than k with xor on it equal to a given x. It's obvious such a query in the array a is equal to a query "how many numbers on the prefix with length k are equal to a given x" in the array p.

If we need to perform a query of the first type it's need to recalculate the array p. Suppose it's required to change value at the position i in a to x. Let's understand how the array p changes. For any $j < i$ $p[j]$ doesn't change since the i-th element doesn't affect to the prefixes before position i. In the same time for any $j \ge i$ old value of $a[i]$ doesn't affect on $p[j]$ anymore, so it's need to "cancel" its contribution in $p[j]$. Since $x \oplus x = 0$ for any x we can do $p[j] = p[j] \oplus a[i]$, thereby saying that an old value of $a[i]$ is not affecting on $p[j]$ anymore. After that do $p[j] = p[j] \oplus x$ meaning we add x to $p[j]$. Therefore, for changing value at position i in array a it's need to do $p[j] = p[j] \oplus c$ for each $j \ge i$, where $c = x \oplus a[i]$.

So queries became as follows:

1. Given i and x, do $p[j] = p[j] \oplus c$ for each $j \ge i$, where $c = a[i] \oplus x$.
2. Find the amount of such i's that $i \le k$ and $p[i] = x$ with given k and x.

It can be done using sqrt-decomposition. Split the array p into blocks with B = length of the each block. It means elements of the array with indexes $[0; B-1]$ belong to the block with index 0, elements with indexes $[B; 2B-1]$ belong to the block with index 1 and so on, elements with indexes $[\frac{N-1}{B} \cdot B, N-1]$ belong to the last block (it can contain less than B elements). It's obvious there are $O(\frac{N}{B})$ such blocks. It's easy to see any index i belongs to the block with index $\frac{i}{B}$. For each block i store an additional value $t[i]$ meaning each value inside this block is xorred with $t[i]$ ($t[i] = 0$ initially), i.e equal to $a[j] \oplus t[i]$. In other words, for each index j of the array, the actual value of j-th prefix xor sum is $p[j] \oplus t[\frac{j}{B}]$. Also each block stores an array $freq[i][j]$ meaning how many values of p equal to j belong to it. Before performing the queries calculate the array freq, just increasing $freq[\frac{i}{B}][p[i]]$ by 1 for each i.

Now suppose query of the first type comes. Suppose index i belongs to the $k = \frac{i}{B}$-th block. Set c to $a[i] \oplus x$ and $a[i]$ to x. Then for any index j after i in the k-th block $p[j]$ should be changed to $p[j] \oplus c$. Just iterate over all such j's and decrease $freq[k][p[j]]$ by 1, increase $freq[k][p[j] \oplus c]$ by 1 and do $p[j] = p[j] \oplus c$. In this way we updated the information in the k-th block in $O(B)$ time. For any block with index $p > k$ we can't update each index independently because it's slow, but we can simply change $t[p]$ to $t[p] \oplus c$ "promising" to xor it with c after, when answering the queries of the second type.

For answering a query suppose i belongs to the $k = \frac{i}{B}$-th block. Then it's need to check each index j before i inside this block. Just iterate over j from the beginning of the k-th block to i and increase the answer if $p[j] \oplus t[\frac{j}{B}]$ is equal to x from the query. For each block p before k, we know the frequency of each value inside it. So we can just add to the answer $freq[p][x]$, but remembering about extra xors it turns into $freq[p][x \oplus t[p]]$.

Follow this question
By Email:
You are not subscribed to this question.
subscribe me
(you can adjust your notification settings on your profile)
By RSS:
Answers
Answers and Comments

Question tags:
editorial ×14,198
dec17 ×178
chefexq ×27
question asked: 13 Dec '17, 03:1
question was seen: 3,449 times
last updated: 13 Jan, 12:26

Related questions
CPLAY - Editorial
Unofficial editorials December L
challengr Part 2
CHEFFIB - Editorial
CHEFHAM - Editorial
REDBLUE - Editorial
CHEFEXQ - Unofficial Editorial
CHEFUNI - Editorial
DS = QUAD TREE Unofficial editor
December Long Challenge Part 3
Video Editorials for December Long
Unofficial editorial December l
challenge

In each query we iterate over $O(\frac{N}{B})$ blocks and one whole block taking $O(B)$ time for it. So each query takes $O(B+\frac{N}{B})$ time. Taking $B=\sqrt{N}$ leads to the optimal $O(\sqrt{N})$ time per query, so the array should be splitted into blocks of size around $\sqrt{N}$. Total time complexity is $O(Q\sqrt{N})$.

AUTHOR'S AND TESTER'S SOLUTIONS:

Author's solution can be found here.
Tester's solution can be found here.

dec17 editorial chefexq

This question is marked "community wiki".

edited 21 Dec '17, 02:42

asked 13 Dec '17, 03:11
6★ kefaa
[11]●2●8
accept rate: 0%

The latex seems to be messed up for no reason for this post. Asked @admin to give it a look.

5★ vijju123 ♦ (14 Dec '17, 20:27)

6 Answers:

oldest answers newest answers popular answers

1

I'm a bit confused. When answering the query 2; if you do not update the frequency table for the updated value. Won't they have invalid values, am i missing something? When we performed operation 1, we did not update frequency values of the remaining blocks. Won't the value inside them, which has count of xors from previous values become invalid?

link | award points

answered 16 Dec '17, 16:56

3★ aman935
[120]●1
accept rate: 0%

1

no,for "other blocks",you have to maintain a lazy array which will be keeping the track that before answering a query ,you are supposed to look for its update in the lazy array and count your no of required values accordingly...

link | award points

answered 17 Dec '17, 18:50

2★ namanjain007
[11]●1
accept rate: 0%

1

Didn't know that we could allocate upto 2 gb memory. thats why I didn't think of sqareroot decomposition with storing frequency.

link | award points

answered 17 Dec '17, 20:42
6★ praveenkumar12
[20]●1
accept rate: 0%

0

Over Here

So queries became as follows:

Given i and x, do p[j]=p[j] ⊕ c for each j≥i, where c = p[i] ⊕ x

Find the amount of such i’s that i≤k and p[i]=x with given k and x.

shouldn't c = a[i] ⊕ x and not p[i] ⊕ x

And one more question: wouldn't making the freq 2D array exceed the memory limit ?

link | award points

edited 18 Dec '17, 11:45

answered 18 Dec '17, 11:37
2★ prasadc8897
[1]●1
accept rate: 0%

c = a[i] xor x, thanks. If you are afraid of big array, you can use hashtable or map instead. It may lead to extra logN in complexity but if you write it carefully it can still easily pass.

6★ kefaa (21 Dec '17, 02:48)

0

why we can't use segmenttree for this problem?? m getting TLE in last subtask ? thanku inadvance!

link | award points

answered 26 Dec '17, 14:20

3★ mritunjay07
[11]●1
accept rate: 0%

0

Why this code is giving WA on a single test case of subtask 4 : https://www.codechef.com/viewsolution/16702764

link | award points

answered 30 Dec '17, 17:59

3★ jain_0709
[1]
accept rate: 0%

Your answer

[hide preview]

community wiki:

Preview

Answer the question

About CodeChef | About Directi | CEO's Corner
CodeChef Campus Chapters | CodeChef For Schools | Contact Us