Beautiful Subarray

Brute Force

Find all the subarray which is beautiful.
Beautiful subarray has all its element equal 0.

Eg: 4 3 1 2 4

4 = 100	3 = 011
4^3 = 111	3^1 = 010
4^3^1 = 110	3^1^2 = 000
4^3^1^2 = 100	↓
4^3^1^2^4 = 0000	subarray ②
↓	3^1^2^4 = 100
subarray ①	

1 = 001	2 = 010	4 = 100
1^2 = 011	2^4 = 110	
1^2^4 = 111		

There are 2 beautiful subarray.

	4	3	1	
4	1	0	0	
3	0	1	1	→ 1st operation Change to 0
1	0	0	1	→ 2nd operation Change to 0
2	0	1	0	
4	1	0	0	
	0	0	0	

3rd operation Change to 0

optimal solution

dict {

0: 0000

4: 100

4^3: 111

4^3^1: 110

4^3^1^2: 100

4^3^1^2^4: 0000

}

Observed 4 = 100 & 4^3^1^2 = 100

If 4 was 100, then to get back 100 again 3^1^2 should be 0.

4 ^ 3 ^ 1 ^ 2 = 100

↓ ↓

100 0

4	1	0	0	
3	0	1	1	cancel out
1	0	0	1	cancel out
2	0	1	0	

	4	1	0	0
3^1^2	0	0	0	# beautiful subarray
	1	0	0	

∴ If an XOR value has appeared before, that means a beautiful subarray with zeros is present