# 1803. Count Pairs With XOR in a Range

## Description

Given a **(0-indexed)** integer array `nums` and two integers `low` and `high`, return *the number of* ***nice pairs***.

A **nice pair** is a pair `(i, j)` where `0 <= i < j < nums.length` and `low <= (nums[i] XOR nums[j]) <= high`.

### Example 1:

```
Input: nums = [1,4,2,7], low = 2, high = 6
Output: 6
Explanation: All nice pairs (i, j) are as follows:
    - (0, 1): nums[0] XOR nums[1] = 5
    - (0, 2): nums[0] XOR nums[2] = 3
    - (0, 3): nums[0] XOR nums[3] = 6
    - (1, 2): nums[1] XOR nums[2] = 6
    - (1, 3): nums[1] XOR nums[3] = 3
    - (2, 3): nums[2] XOR nums[3] = 5
```

### Example 2:

```
Input: nums = [9,8,4,2,1], low = 5, high = 14
Output: 8
Explanation: All nice pairs (i, j) are as follows:
    - (0, 2): nums[0] XOR nums[2] = 13
    - (0, 3): nums[0] XOR nums[3] = 11
    - (0, 4): nums[0] XOR nums[4] = 8
    - (1, 2): nums[1] XOR nums[2] = 12
    - (1, 3): nums[1] XOR nums[3] = 10
    - (1, 4): nums[1] XOR nums[4] = 9
    - (2, 3): nums[2] XOR nums[3] = 6
    - (2, 4): nums[2] XOR nums[4] = 5
```

### Constraints:

- `1 <= nums.length <= 2 * 10^4`
- `1 <= nums[i] <= 2 * 10^4`
- `1 <= low <= high <= 2 * 10^4`