You're given an array of n integer.

Return the number of pairs(a, b) such that array[a] & array[b] = 0(& symbol means bitwise and).

*Note:*  
Pairs (a, b) and (b, a) are considered to be different.

**Example**

For n = 5 and array = [41, 47, 34, 40, 29] the output should be 2.

* **[input] integer n**
  + n < 2000
* **[input] array.integer array**
  + array[i] < 5000
* **[output] integer**

<https://codefights.com/challenge/pYtYvTEkW45x8NwMh>

--ACEPTADO--

int bitwiseandzero(int n, std::vector< int > array) {

int cont = 0;

for(int i = 0; i < n - 1; i++){

for(int j = i+1; j < n; j++)

{

if((array[i] & array[j]) == 0){

cont++;

}

if((array[j] & array[i]) == 0) {

cont++;

}

}

}

return cont;

}