Name :- Asmit Prabhakar

Uid :- 23BCS10063

Class KRG 2B 2'0

Q. we define f(x,y) as number of different corresponding bits in the binary representation of x and y. For example f(2,7) = 2, since the binary representation of 2 and 7 are 010 and 111. The first and third bit differs by So, f(2,7) = 2.

Test case :-

Input : [1,3,5]

output = 8

Input = 2,3

output = 2.

Approach :-

i) First we will take XOR of two numbers.
ii) Then we we will check the set bits (val > 0) of it and store it in count, and
iii) Right shift the value (comes from XOR) by by 1.

Brute force :-

```
f(x,y){
    int num = x^y;
    while(num>0) {
        count+ = (num&1);
        num >> 1;
    }
    return count;
}
main(){
    for(int i=0; i<n; i++){
        for(int j = i+1; j<n; j++){
```

int value += ~~f(i,j)~~ f(num[i], num[j]) % mod;

}
cout << value;

}

→ optimal approach :-

long long do(vector<int> & nums) {
long long mod = 1e9+7;
long long n = nums.size();
long long ans = 0;
for(int b=0; b<32; b++) {
long long count=0;
for(int i=0; i<n; i++){
if(nums[i] & (i<<b)) {
count++;
}
long long zeroes = n-zeroes;
ans = (ans + ((count+zeroes) % mod) * 2) % mod;
}
}

Time complexity :- O(n x 32)
= O(n).