**Sum of Products**

[bit](http://www.practice.geeksforgeeks.org/tag-page.php?tag=bit&isCmp=0)[maths](http://www.practice.geeksforgeeks.org/tag-page.php?tag=maths&isCmp=0)

Given an array **A[ ]** of **N** integers, calculate the sum of "**A[i] & A[j]**" of all the pairs formed by the given array, where**&** is the **bitwise AND**operator.

**Input**  
The first line of input contains an integer **T**denoting the number of test cases. Then **T** test cases follow.   
The first line of each test case contains a positve integer **N**, denoting the length of the array **A[ ]**.  
The second line of each test case contains a **N**space seprated positve integers, denoting the elements of the array **A[ ]**.

**Output**  
Print out the sum of products of all pairs formed by the array.

**Constraints**  
1 <= **T** <= 100  
2 <= **N** <=30  
0 <= **A[ ]** <= 100

**Examples**

**Input**  
3  
3  
5 10 15  
4  
10 20 30 40  
5  
20 16 32 50 64

**Output**  
15  
46  
80

**Explanation:**  
For the above test case   
Required Value = (5 & 10) + (5 & 15) + (10 & 15)   
                             = 0 + 5 + 10   
                             = 15

**Expected Complexity**  
**Time**: O(N)

\*\*For More Examples Use Expected Output\*\*

<http://www.practice.geeksforgeeks.org/problem-page.php?pid=944>

#include <iostream>

#include <stdio.h>

using namespace std;

int main() {

int t;

scanf("%d", &t);

while(t--) {

int n;

scanf("%d", &n);

int A[n];

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

scanf("%d", &A[i]);

}

int sum =0;

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

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

sum += (A[i] & A[j]);

}

}

cout << sum << endl;

}

return 0;

}