C. Make It Beautiful

time limit per test: 2 seconds
memory limit per test: 512 megabytes

You are given an array a of n integers. We define the beauty of a number x to be the number of 1 bits in its binary representation. We define the beauty of an array to be the sum of beauties of the numbers it contains.

In one operation, you can select an index i $(1 \leq i \leq n)$ and increase a_i by 1.

Find the maximum beauty of the array after doing **at most** k operations.

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \leq t \leq 5000$). The description of the test cases follows.

The first line of each test case contains two integers n and k ($1 \leq n \leq 5000$, $0 \leq k \leq 10^{18}$) — the length of the array and the maximal number of operations.

The second line of each test case contains n integers $a_1, a_2, \ldots a_n$ ($0 \leq a_i \leq 10^9$) —denoting the array a.

It is guaranteed that the sum of n over all test cases does not exceed 5000.

Output

For each test case, output a single integer, the maximum beauty after at most k operations.

Example

input Copy

```
5
5 2
0 1 7 2 4
5 3
0 1 7 2 4
1 1
3
3 0
2 0 3
1 100000000000
0
```

output Copy

```
8
9
2
3
36
```

Note

In the first test case, $a = [0, 1, 7, 2, 4]$.

- apply the first operation at $i = 1$, the new array is $a = [1, 1, 7, 2, 4]$
- apply the second operation at $i = 4$, the new array is $a = [1, 1, 7, 3, 4]$

The beauty of this array is $1 + 1 + 3 + 2 + 1 = 8$. One of the other valid solutions with the same beauty is $[0, 1, 7, 3, 5]$.
In the third test case, $a = [3]$. Since you are not required to use exactly k operations, it is optimal to do none.