Problem D - Useless Hash

Let's consider a special kind of hash on n dimensional boolean vectors. A boolean vector is a sequence of n 0/1 characters. We want to find a weight function on each dimension: $(w_1, w_2, \cdots, w_n)$ so we can define the hash function h on $s = s_1 s_2 \cdots s_n$ by

$$h(s) = w_1 s_1 + w_2 s_2 + \cdots + w_n s_n$$

Moreover, we add some contraint to this hash function:

1. Each w_i is a nonnegative power of some fixed positive integer X.
2. For any two boolean vector a, b, if they have different number of 1 entries, then $h(a) \neq h(b)$.
3. On some concerned pairs $(x_1, y_1), \ldots, (x_m, y_m)$ we want that $h(x_i) \neq h(y_i)$ whenever $x_i \neq y_i$ for any $i = 1, 2, \cdots, m$.

Clearly setting $w_i = 2^{i-1}$ will be a valid hash function. But now we want to minimize $w_1 + w_2 + \cdots + w_n$, can you find such a hash function?

Input

First line contains an integer T ($1 \leq T \leq 100$), denoting the number of test cases.

For each test case, first line contains two integers n, m ($1 \leq n \leq 10, 1 \leq m \leq 1000$) separated by a whitespace. Each of the next m lines contains two binary strings x_i, y_i of length n. It is guaranteed that $x_i \neq y_i$.

Output

For each test cases, please output the minimum possible sum of all weights over all valid hash functions.

Sample Input

```
3
3 3
001 010
100 001
010 100
3 0
3 1
011 101
```

Sample Output

```
7
3
5
```