

Knapsack

Input file: **standard input**
Output file: **standard output**
Time limit: 4 seconds
Memory limit: 1024 megabytes

There are n groups of items, where the i -th group contains a_i items, and each item has a weight of 2^{b_i} . There are also m knapsacks, each with a capacity of k . Calculate the smallest k such that it is possible to put all $\sum_{i=1}^n a_i$ items into the knapsacks, and the total weight of the items in each knapsack does not exceed k .

Note that each item should be placed into exactly one knapsack. One knapsack may contain items from different groups, and the items from the same group may go into different knapsacks.

Input

There are multiple test cases. The first line of the input contains an integer T ($1 \leq T \leq 10^4$) indicating the number of test cases. For each test case:

The first line contains two integers n and m ($1 \leq n \leq 2 \times 10^5$, $1 \leq m \leq 10^9$) indicating the number of groups of items and the number of knapsacks.

For the following n lines, the i -th line contains two integers a_i and b_i ($1 \leq a_i \leq 10^9$, $0 \leq b_i \leq 10^9$), where a_i is the number of items in the i -th group, and 2^{b_i} is the weight of each item in the i -th group.

It's guaranteed that the sum of n of all test cases does not exceed 2×10^5 .

Output

For each test case, output one line containing one integer indicating the answer. As the answer may be large, output it modulo 998 244 353. Note that the modulus here is only to simplify the output, and you need to minimize the answer before the modulus.

Example

standard input	standard output
2	10
5 4	628956724
3 0	
2 3	
3 1	
1 3	
2 1	
2 20250427	
1000000000 1000000000	
114514 1919810	